creative commons
COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

공학석사 학위논문

3D-CNN을 이용한 효율적인 손 제스처 인식 방법에 관한 연구

3D-Convolutional Neural Network for Efficient Hand Gesture Recognition

2017 년 2 월

서울대학교 대학원

전기컴퓨터공학부

이 진 원

3D-CNN을 이용한 효율적인 손 제스처 인식 방법에 관한 연구

3D-Convolutional Neural Network for Efficient Hand Gesture Recognition

지도 교수 이 혁 재

이 논문을 공학석사 학위논문으로 제출함
2017 년 2 월

서울대학교 대학원
전기 컴퓨터 공학부
이 진 원

이진원의 공학석사 학위논문을 인준함
2017 년 2 월

위 원 장 _____ (인)

부위원장 _____ (인)

위 원 _____ (인)

초 록

손 제스처 인식(Hand gesture recognition) 기술은 사람이 손을 이용하여 미리 정해진 동작을 했을 때, 그것이 어떤 동작인지를 인식하는 기술을 말한다. 이러한 인식 기술은 제스처의 특성상 직접적인 접촉을 필요로 하지 않기 때문에, 자동차나 모바일 혹은 웨어러블 기기, 가전제품 등에 있어서 효과적인 인터페이스를 제공해줄 수 있는 아주 중요한 기술 중 하나이다. 많은 컴퓨터 비전 알고리즘들이 제스처 인식을 위하여 연구되어 왔고, 그 성능 또한 계속 발전해왔다. 특히, 최근에는 인공신경망과 딥러닝 기술의 발전으로 인하여, 이러한 알고리즘들이 기존의 많은 연구들을 뛰어넘는 성과를 보이고 있다. 그러나 손 제스처 인식 기술은 영상 데이터의 처리의 특성상 높은 복잡도와 많은 연산량 그리고 많은 메모리 사용량을 요구하는 반면 많은 기기들의 제한된 연산능력으로 인하여 여전히 도전적인 분야이고, 더 효과적이고 효율적인 기술이 요구되는 분야이다.

본 논문에서는 연산량과 메모리 사용량을 줄일 수 있는 3차원 컨볼루션 신경망(3D convolutional neural network)을 이용한 새로운 구조의 인공신경망을 제안한다. 영상 데이터는 시간상의 연속된 움직임을 담고 있기 때문에 프레임 단위로 보았을 때 이웃한 프레임 간에는 움직임이 있는 부분을 제외하고는 큰 차이를 보이지 않는 것이 특징이다. 이를 이용하여 카메라를 통해 입력되는 RGB 영상의 프레임 간 차이를 이용한 차영상을 인공신경망의 입력으로 사용하는 방법을 제안한다. 또한, 이미지 분류에서 비교적 적은 수의 학습 변수(Weight parameter)를 이용하여 뛰어난 성능을 보였던 구조 중 하나인

인셉션(Inception) 구조를 영상 데이터 즉 3차원 데이터에 적용 가능하도록 확장하고 내부 필터를 작은 필터들의 결합으로 분해한 구조의 새로운 신경망을 제안한다. 마지막으로, 인공 신경망의 마지막 단에 전결합층(Fully-connected layer) 대신 3차원 글로벌 평균 풀링(3D global average pooling) 사용을 제안하여 학습 변수의 양을 줄이고, 제안된 신경망이 시간과 공간상의 변화에 잘 대응할 수 있도록 하였다.

제안된 인공 신경망 구조를 사용할 경우, 일반적인 구조의 3차원 컨볼루션 신경망을 사용하는 구조와 비교하였을 때, 인식률의 저하 없이 학습 변수를 저장하기 위해 필요한 메모리 사용량을 99% 이상 줄일 수 있다. 또한 연산에서 가장 많은 비용이 발생하는 곱셈 연산의 경우 역시 인식률의 저하 없이 일반 구조의 3D 컨볼루션 신경망 대비 약 95%의 연산량을 줄일 수 있다.

주요어 : 손 제스처 인식, 3D 컨볼루션 신경망, 차영상, 인셉션, 글로벌 평균 풀링

학 번 : 2002-21592

목 차

표 목차

그림 목차

제 1 장 서 론

1.1 연구의 배경 및 내용

손 제스처 인식(Hand gesture recognition) 기술은 인간과 기계의 의사소통을 위한 하나의 수단이 되는 기술 중 하나로 자동차나 모바일 혹은 웨어러블 기기, 또는 가전제품 등 다양한 기기에 적용될 수 있는 중요한 기술이다. 예를 들어 자동차의 경우, 제스처를 이용하여 에어컨이나 오디오와 같은 장치의 조작이 가능하게 하면 그만큼 운전에 집중할 수 있게 되어 운전자의 안전과 편리성을 더 높일 수 있으며, 키보드나 터치스크린이 없는 다양한 모바일이나 웨어러블 기기들의 제어에도 제스처 인식 기술이 효과적인 인터페이스가 될 수 있다. 이 밖에도 이러한 제스처 인식 기술을 이용한 다양한 연구들이 이루어지고 있는데, 그 대표적인 분야로 수화 인식이나 거짓말 탐지 혹은 로봇 제어 등을 들 수 있다.

일반적인 제스처의 경우 특정한 시간과 공간 상에서 변화하는 손의 움직임을 인식하는 기술이기 때문에, 사물 인식과 같이 정지된 사진에서의 특정 물체를 인식하는 기술에 비하여 어렵고, 연산량 또한 많이 필요로 한다. 컴퓨터 영상 인식 분야에서는 오래전부터 이러한 제스처 인식의 정확도를 높이기 위한 다양한 연구를 해왔고, 그 성능

또한 지속적으로 증가해왔다. 그러나 한편으로는 이러한 제스처 인식이 응용되는 많은 분야에서 실시간 처리를 필요로 하고 있으며, 모바일 기기나 웨어러블 기기와 같이 연산 능력에 제한이 있는 곳에서의 수요 또한 커지고 있기 때문에 많은 연산량과 메모리 사용량은 제스처 인식에 있어서 큰 도전이 되고 있다.

한편, 최근 딥러닝 기술의 발전으로 인하여 다양한 인식 기술 분야에서 딥러닝 알고리즘을 이용한 뛰어난 결과들이 많이 발표되고 있다. 대표적인 알고리즘으로 컨볼루션 신경망(Convolutional neural network)과 순환 신경망(Recurrent neural network)을 들 수 있는데, 컨볼루션 신경망의 경우에는 시각적 물체 인식과 같은 분야에서 탁월한 업적을 달성하고 있으며, 순환 신경망의 경우에는 자연어 처리 및 기계 번역과 같은 시계열 데이터 처리의 정확성 향상에서 뛰어난 성능을 보여주고 있다.

사람의 제스처나 행동 인식 분야에서도 이러한 딥러닝 알고리즘을 이용한 기술들이 많이 연구되고 있다. 제스처의 경우 시간과 공간상의 정보를 모두 고려해야 하기 때문에 기존의 컨볼루션 신경망에 시간상의 정보를 고려할 수 있도록 차원을 확장한 구조의 3차원 컨볼루션 신경망(3D-convolutional neural network)이나[1-3], 기존의 컨볼루션 신경망에 순환 신경망을 더하여 공간적인 특징은 컨볼루션 신경망으로, 시간적인 특징은 순환 신경망으로 찾아내는 구조의 신경망[4, 5], 혹은 이 둘을 합쳐놓은 구조의 신경망[8, 9] 등이 제안되어 왔다. 일반적으로 3차원 컨볼루션 신경망의 경우 비교적 짧은

시간에 일어나는 행동에 대한 인식에 좋은 결과를 보인 반면, 컨볼루션 신경망과 순환 신경망을 결합한 형태의 구조는 상대적으로 긴 시간에 걸쳐 일어나는 행동에 대한 인식에서 좋은 결과를 보인다는 것이 기존의 연구를 통해 알려져 있다. 손 제스처 인식은 대부분의 경우에, 짧은 시간 동안 한정된 공간에서 일어나는 행동을 판별하는 기술이기 때문에, 비교적 단순한 형태의 3차원 컨볼루션 신경망을 사용하는 것이 좋은 선택이 될 수 있다.

한편, 인식 기술에 있어서 딥러닝의 발달로 인하여 다양한 분야에서 훌륭한 성능을 보여주고 있지만, 더 높은 성능을 위하여 신경망의 구조가 더욱 복잡해지고 그 크기가 커짐에 따라서, 비슷한 성능을 유지하면서 기존 신경망 구조를 최적화하는 문제가 점점 더 중요해지고 있다. 예를 들어 [3]에서 제안된 3차원 컨볼루션 신경망의 경우 가장 아래쪽 4개 층에만 학습 해야하는 파라미터가 290만 개 존재하고 이것은 많은 연산량과 함께 이 파라미터를 저장하기 위한 많은 메모리 공간이 필요함을 의미한다. 본 논문에서는 이러한 문제를 해결하기 위하여 단순한 3차원 컨볼루션 신경망을 최적화하기 위한 몇가지 기술을 제안한다.

첫 번째로, 영상 데이터의 특징을 이용하여 기존의 영상 입력이 아닌, 차영상을 입력으로 사용하여 연산량을 줄이는 방법을 제안한다. 두 번째로, 기존 컨볼루션 신경망에서 적은 파라미터로 좋은 성능을 보인 인셉션 구조를 3차원 구조에 맞게 확장하고 인셉션 내부에서 사용되는 필터를 더 작은 필터들로 분할하여 파라미터 수를 줄임으로써

연산량을 감소시킬 수 있는 방법을 제안한다. 마지막으로 3차원 글로벌 평균 풀링을 사용하여, 컨볼루션 신경망의 마지막 단인 전결합층의 많은 파라미터 수를 획기적으로 감소시킴으로써 파라미터 저장을 위한 메모리 사용량 또한 감소시킬 수 있는 방법을 제안한다.

1.2 논문 구성

본 논문은 다음과 같이 구성되어 있다. 2 장에서는 컨볼루션 신경망과 그 특징 및 이를 이용한 행동 인식 연구에 대하여 설명한다. 1절에서 일반적인 2차원 컨볼루션 신경망에 대하여 먼저 소개하고 2절에서 이를 시간 축으로 한 차원 확장한 3차원 컨볼루션 신경망에 대하여 설명한 후, 3절에서 이러한 컨볼루션 신경망을 이용한 사람의 행동 인식 분야의 기존 논문에 대하여 간략히 소개하도록 한다. 3 장에서는 본 논문에서 제안한 세 가지 방법에 대하여 설명한다. 1절에서는 차영상을 입력으로 이용하는 방법을 설명하고 2절에서는 인셉션 구조를 3차원으로 확장한 신경망 구조에 대하여 설명한다. 그리고 마지막 3절에서는 3차원 글로벌 평균 풀링에 대하여 소개하고 이의 특징에 대하여도 설명한다. 4장에서는 제안한 방법들에 대한 성능 비교 및 연산량과 필요한 메모리 사용량에 대하여 비교 분석하고, 학습 과정이 잘 이루어졌는지 그리고 제안된 신경망을 사용하여 추가적으로 얻을 수 있는 정보들이 무엇인지에 대하여 설명한다. 마지막 5장에서는 본 논문에서 제안하는 방법과 결과들에 대하여 정리하고 결론을 맺는다.

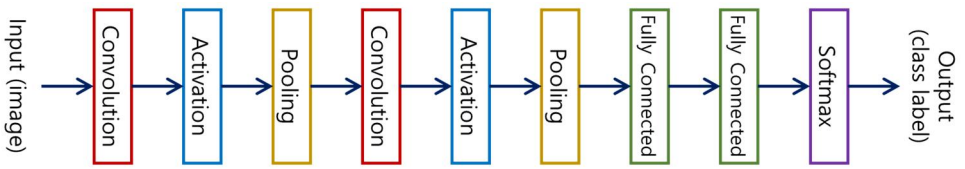
제 2 장 컨볼루션 신경망과 관련 연구

2.1 2차원 컨볼루션 신경망

컨볼루션 신경망(Convolutional neural network)는 CNN이라고 불리우며 주로 이미지 인식 분야를 기반으로 한 여러 응용 분야에서 탁월한 성능을 보여주고 있는 인공 신경망이다. 컨볼루션 신경망은 생물의 뇌에 있는 시각 피질(visual cortex)에 대한 신경과학적 지식으로부터 힌트를 얻어 만들어졌다. 시각 피질에는 눈과 같은 감각기관을 통해 입력을 받는 감수 영역(receptive field)이 국소적인 특징을 띠고 특정 패턴에만 반응하는 선택적 반응성을 보이는 세포들이 존재하는데, 이러한 특징을 모방하여 컨볼루션 신경망이 만들어졌다. 최초의 컨볼루션 신경망은 네오코그니트론(Neo-Cognitron)[17]으로 패턴의 위치에 민감하게 반응하는 단순 세포와 패턴의 위치 보다는 연결된 세포들의 반응 유무에 더 민감하게 반응하는 복잡 세포를 모형화한 구조를 이용하여 이를 패턴인식에 적용하였다. 그러나 컨볼루션 신경망이 본격적으로 사용되기 시작한 것은 LeNet[18]이 그 시초이다. LeNet은 컨볼루션 신경망을 역전파법(back-propagation)과 경사하강법(gradient decent)으로 파라미터를 최적화한 신경망으로 문자 인식에서 높은 성능을 보여주었다. 현재 ImageNet 대회와 같은

사물 분류 분야에서 컨볼루션 신경망은 놀라운 성능을 보여주고 있으며, 가장 널리 사용되는 인공 신경망 중 하나이다.

[그림 2-1]에 단순한 컨볼루션 신경망의 일반적인 구조를 나타내었다.



[그림 2-1] 컨볼루션 신경망의 일반적인 구조

입력 이미지로부터 출력단까지 진행하면서, 컨볼루션층(convolution layer), 활성화 함수(activation function), 풀링층(pooling layer)의 구조가 반복되고 마지막에 전결합층(fully connected layer)이 반복된 후에 소프트맥스(softmax) 함수를 통하여 입력 이미지가 어떤 클래스에 해당하는 지를 판단하게 되는 구조이다. 각 계층의 특징 및 역할은 다음과 같다.

2.1.1 컨볼루션층(Convolution layer)

컨볼루션층에서는 이전 단의 입력을 받아 필터와의 컨볼루션 연산을 수행한다. 이미지와 필터 사이에 정의되는 컨볼루션 연산은 다음과 같다.

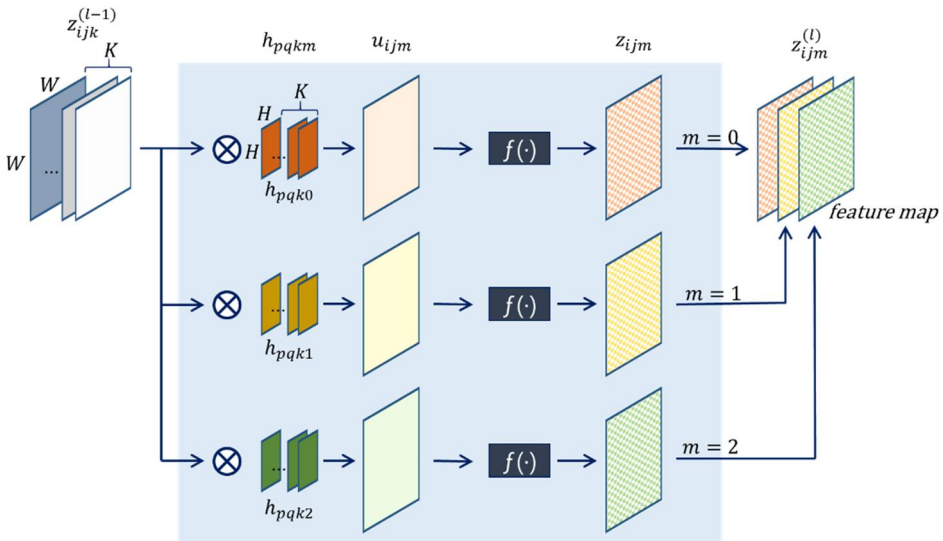
$$u_{ij} = \sum_{p=0}^{H-1} \sum_{q=0}^{H-1} x_{i+p,j+q} h_{pq} \qquad (1)$$

필터의 크기는 H x H 이고, h_{pq} 는 필터의 픽셀 값을, $x_{i+p,j+q}$ 는 이미지의 픽셀 값을 의미한다. 컨볼루션 층에서는 필터가 이미지의 가로 세로 방향으로 이동(sliding)하면서 이러한 컨볼루션 연산을 반복하게 되는데, 이 때 한 번 이동시에 얼마나 건너 뛰면서 이동할 것인지(stride), 그리고 이미지 가장자리 부분의 컨볼루션 연산을 위하여 이미지 바깥 영역에 특정 값을 채우고 연산을 진행할 것인지(padding)에 대한 선택에 따라 다양한 방식이 존재한다.

컨볼루션 신경망에서는 이러한 컨볼루션 연산이 이미지 한 장이 아니라 여러 개의 채널을 갖는 이미지에 대해서 이루어지게 되는데, 채널 수가 K인 이미지의 각 픽셀은 K개의 값을 갖는다. 그리고 입력층에서는 대부분 RGB와 같은 3개의 채널을 갖는 즉 K=3인 이미지를 입력으로 받지만 중간층으로 가게 되면 대부분 그 이상의 채널 수를 갖게되며, 이러한 경우 중간층의 입력으로 들어오는 다채널 이미지를 특성맵(feature map)이라고 부른다. 이것은 컨볼루션 층을 통과하면서, 필터를 통해 입력 이미지의 특정한 패턴 혹은 특성이 추출되기 때문이다. 이미지 혹은 특성맵의 가로, 세로의 화소 수가 $W \times W$이고 채널 수가 K인 경우, $l-1$ 번째 층의 입력을 받아 l번째 특성맵을 만들어내는 활성화 함수(f)를 포함하는 컨볼루션층의 구조를 **[그림 2-2]**에

나타내었다. 필터는 이미지와 같은 채널 수 K를 가지며, 그 크기는 H × H이다. 또한 그림에 나타난 필터는 3개이며, 이것이 다음 층의 입력으로 사용될 특성맵의 채널 수를 결정한다. 이 그림에서 컨볼루션 연산은 다음과 같은 식으로 표현할 수 있다. 식 마지막 부분에 더해진 b_{ijm}은 바이어스를 의미한다.

$$u_{ijm} = \sum_{k=0}^{K-1} \sum_{p=0}^{H-1} \sum_{q=0}^{H-1} z_{i+p,j+q,k}^{(l-1)} h_{pqkm} + b_{ijm} \quad (2)$$



[그림 2-2] 2차원 컨볼루션층의 구조

2.1.2 활성화 함수(Activation function)

활성화 함수는 **[그림 2-2]**에서 f로 표시된 부분으로 비선형 함수를 말한다.

$$z_{ijm} = f(u_{ijm}) \qquad (3)$$

신경망을 다층(multi-layer)으로 구성하는 이유는 더욱 복잡한 문제를 풀 수 있도록 하기 위함인데, 컨볼루션 연산이 선형 연산이기 때문에, 신경망의 다층 효과를 보려면 활성화 함수와 같은 비선형 함수를 반드시 사용해야 한다. 과거에는 활성화 함수로 시그모이드(sigmoid) 함수가 많이 사용되었으나, 그래디언트 소멸 문제(vanishing gradient problem)로 인하여, 최근에는 ReLU(Rectified Linear Unit)가 많이 사용되고 있다. ReLU는 식 (4)에서 보는 것과 같이 음수의 입력을 받으면 0을 출력하고 양수의 입력은 그 값을 그대로 출력하는 함수이다.

$$f(x) = \max(0, x) \qquad (4)$$

이 함수를 사용하면, 그래디언트 소멸 문제를 해결할 수 있다. 또한 ReLU를 사용하면 입력이 양수인 경우는 ReLU도 선형 함수의 형태이므로, 입력에 따라서 달라지는 선형 함수들의 결합으로 신경망이

이루어지게 된다는 특징이 있다.

2.1.3 풀링층(Pooling layer)

풀링층은 보통 컨볼루션 층과 활성화 함수에 이어서 배치된다. 풀링층은 특정한 패턴의 위치보다는 연결된 세포들의 반응 유무에 영향을 받는 복잡 세포를 모형화한 것으로 이미지 내에서 특정 패턴의 위치가 변하여도 같은 결과를 낼 수 있도록 해주는 역할을 한다. 크기가 $W \times W \times K$인 입력 이미지에서 $H \times H$의 정사각형을 잡아 그 영역에 포함되는 픽셀의 집합을 P_{ij} 라고 하였을 때, H^2 개의 픽셀 값을 이용하여 K개의 채널마다 독립적으로 하나의 픽셀값 u_{ijk}를 구하는 과정이 풀링층에서 일어난다. 대표적인 풀링 방법으로는 최대 풀링(max pooling)과 평균 풀링(average pooling)이 있는데, 최대 풀링은 식 (5)와 같이 픽셀값 중 최댓값을 고르는 것이고 평균 풀링은 식 (6)과 같이 픽셀값의 평균값을 계산하여 출력으로 내보내는 것이다.

$$u_{ijk} = \max_{(p,q)\in P_{ij}} z_{pqk} \qquad (5)$$

$$u_{ijk} = \frac{1}{H^2} \sum_{(p,q)\in P_{ij}} z_{pqk} \qquad (6)$$

2.1.4 전결합층(Fully connected layer)

전결합층은 일반적으로 컨볼루션 신경망의 맨 마지막에 위치하며, 전결합층에서는 컨볼루션층과 풀링층을 통해 추출된 특성들을 결합하여 어떤 클래스에 해당하는 지를 판단하는 역할을 한다. 전결합층에서는 마지막 특성맵의 모든 픽셀을 벡터화하여 각각의 파라미터를 곱한 값을 최종적으로 합하여 그 중에 가장 큰 값을 갖는 클래스를 결과로 내보내게 되는데, 이 때 각각의 값이 0에서 1 사이의 확률 값으로 나올 수 있도록 소프트맥스 함수를 사용한다. 클래스의 수가 K개라고 할 때 전결합층과 소프트맥스 함수의 식은 각각 아래와 같다.

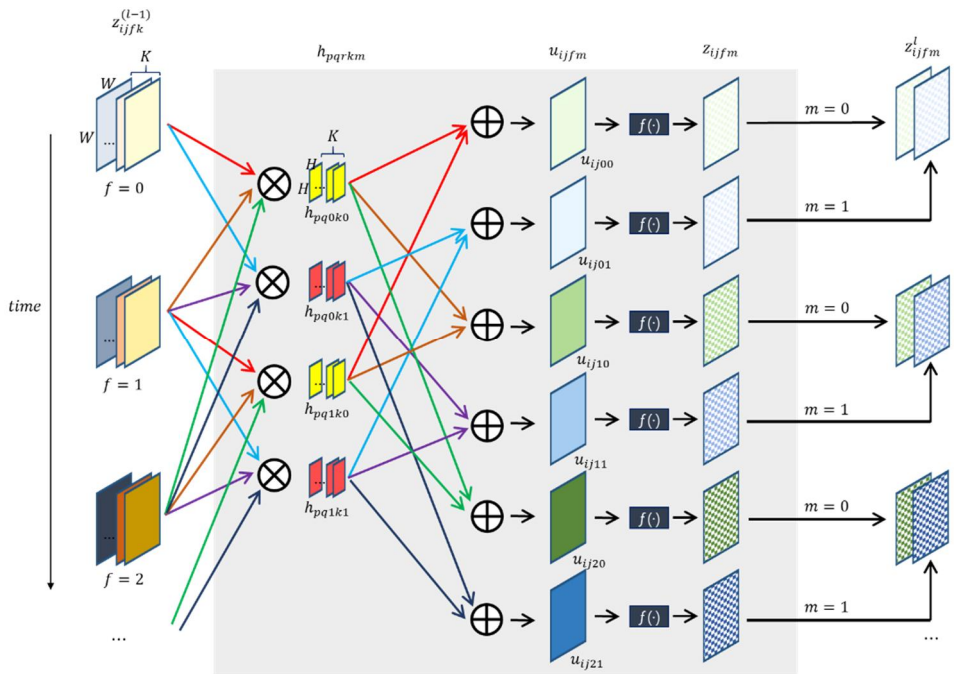
$$u_{jk} = \sum_{i} w_{jik} z_i + b_{jk} \qquad (7)$$

$$z_k = \frac{e^{u_k}}{\sum_{j=1}^{K} e^{u_j}} \qquad (8)$$

2.2 3차원 컨볼루션 신경망

3차원 컨볼루션 신경망은 2.1에서 살펴본 2차원 컨볼루션 신경망을 시간축으로 한 차원 확장시킨 형태의 인공 신경망이다. 2차원 컨볼루션 신경망은 일반적으로 이미지를 입력으로 받아서 그 공간적인 특성을 찾아내어 이미지를 분류하거나, 그와 관련된 응용에 뛰어난 성능을

보이지만, 시간 정보는 다룰 수가 없기 때문에 동영상 데이터는 처리할 수 없다는 한계가 있다. 반면 3차원 컨볼루션 신경망은 컨볼루션 연산과 풀링 연산 등을 시간 성분까지 함께 추가하여 계산하도록 함으로써, 영상 데이터의 특징을 뽑아낼 수 있는 장점이 있다. **[그림 2-3]**에 3차원 컨볼루션층의 구조를 나타내었다.



[그림 2-3] 3차원 컨볼루션층의 구조

컨볼루션 연산이 수행되는 과정에서 데이터의 흐름을 이해하기 쉽게 표현하기 위하여 여러가지 색을 사용하였다. 같은 색을 가지는 화살표는 하나의 컨볼루션 연산을 의미하는데, 3차원 컨볼루션층에서는 4차원

텐서(tensor)간의 연산이 이루어지므로 그림으로 나타내기 용이하도록 연산을 나누어 표현한 것이다. 그림에 나타나있는 컨볼루션 필터의 수는 2개이며, 필터의 크기는 $\mathrm{H} \times \mathrm{H} \times 2(\times \mathrm{K})$ 이고, 연속된 3개의 프레임 데이터를 처리하는 과정을 보여준다.

일반적으로 3차원 컨볼루션은 입력으로 $\mathrm{W} \times \mathrm{W} \times \mathrm{F} \times \mathrm{K}$ 의 영상 데이터를 받는다. 여기서 $\mathrm{W} \times \mathrm{W}$ 는 영상의 해상도를 F 는 한 번에 처리할 프레임 수를, 그리고 K 는 채널 수를 의미한다. 또한 3차원 컨볼루션 연산에 사용되는 필터는 $\mathrm{H} \times \mathrm{H} \times \mathrm{R} \times \mathrm{K}$의 크기를 가지며, 역시 $\mathrm{H} \times \mathrm{H}$ 는 필터의 가로와 세로 크기를, R 은 필터가 한 번에 보고자 하는 프레임 수를, 그리고 K는 채널 수를 의미한다. 2차원 컨볼루션과 마찬가지로 필터가 이미지를 스캔하듯이 움직이면서 컨볼루션 연산을 하는데, 공간상으로 즉 가로와 세로 방향뿐만 아니라 시간축으로도 스트라이드만큼 이동하여 컨볼루션 연산을 수행하게 된다. 이것을 식으로 표현하면 아래와 같이 나타낼 수 있다.

$$u_{ijfm} = \sum_{k=0}^{K-1} \sum_{p=0}^{H-1} \sum_{q=0}^{H-1} \sum_{r=0}^{R-1} z_{i+p,j+q,f+r,k}^{(l-1)} h_{pqrkm} + b_{ijfm} \quad (9)$$

풀링층 또한 시간축으로 한 차원 확장된 형태로, 공간상의 특정 픽셀 값들을 이용해 하나의 값을 만들어 내는 것이 아닌 시간까지 고려하여 3차원 공간 안에 있는 픽셀들의 값을 이용한다. 최대 풀링은 그 값들

중에 최대값을, 평균 풀링은 그 값의 평균을 계산하게 된다. 전결합층에서는 마지막 특성맵에 존재하는 모든 픽셀을 2차원 컨볼루션 신경망과 마찬가지로 벡터화하여 파라미터와의 가중합을 구하게 된다. 클래스를 판단하는 과정도 동일하게 소프트맥스 함수를 이용한다.

이와 같이, 3차원 컨볼루션 신경망은 3차원 필터와 컨볼루션, 풀링 등을 통하여 공간 상의 정보뿐만 아니라 시간 상의 정보도 함께 고려함으로써 영상데이터에 대한 학습이 가능하다는 장점을 가지고 있다. 하지만, 다른 한편으로는 차원이 늘어나게 됨에 따라서, 2차원 컨볼루션 신경망에 비해 한자리 이상 많은 파라미터 수와 연산량이 필요하여 많은 메모리와 높은 연산 처리 능력이 요구된다.

2.3 컨볼루션 신경망을 이용한 행동 인식 연구

제스처 인식이나 이와 유사한 사람의 행동 인식에 관한 연구는 과거부터 활발히 이루어져 왔다. 또한 딥러닝 기술의 발전으로 인하여 이러한 인식 문제를 딥러닝 알고리즘으로 풀어보려는 연구가 최근 많이 이루어지고 있다. 그리고, 이러한 연구의 흐름은 앞에서 얘기한 3차원 컨볼루션 신경망을 사용하는 방법과, 2차원 혹은 3차원 컨볼루션 신경망에 순환 신경망을 결합한 형태의 새로운 신경망을 사용하는 방법으로 크게 나눌 수 있다. 본 절에서는 컨볼루션 신경망을 이용한 제스처나 사람의 행동 인식에 관한 연구에 대하여 살펴보고자 한다.

우선 3차원 컨볼루션 신경망을 이용한 연구에는 [2, 3, 1] 등이

있다. [2]는 사람의 행동을 인식하기 위한 3차원 컨볼루션 신경망을 제안하였다. 60x40의 해상도를 갖는 7 프레임의 영상 데이터를 입력으로 사용하였고, 전처리를 통하여 입력데이터를 그레이 채널, 수평 방향 그래디언트, 수직 방향 그래디언트, 수평 방향 옵티컬 플로우, 수직 방향 옵티컬 플로우의 5개 채널로 확장시킨 다음, 각각에 대하여 3차원 컨볼루션을 적용하였다. 최종단에는 전결합층을 통하여 모든 채널에서 나온 정보를 하나로 묶어서 분류하는 방법을 사용하였다. [3]은 행동 인식을 위하여 깊은 3차원 컨볼루션 신경망을 이용하였는데, 기본 구조는 ImageNet 대회에서 좋은 성능을 보여준 VGG[20]를 3차원으로 확장한 구조와 매우 유사한 구조로 3x3x3 컨볼루션 필터와 2x2x2 최대 풀링의 조합으로 좋은 성능을 보여주었다. [1]은 손 제스처 인식을 위한 3차원 컨볼루션 신경망을 제안하였는데, 이 신경망의 특징은 컨볼루션 신경망을 2개를 사용하여 한 쪽에는 원본 영상 그대로를 다른 한 쪽에는 1/2로 축소한 영상을 입력으로 넣어주고 맨 마지막 전결합층에서 이 두 신경망을 합쳐서 최종적으로 제스처를 판단하는 형태를 가지고 있다는 점이다.

한편, [8]는 2개의 2차원 컨볼루션 신경망 사용하였는데 한 쪽은 영상을 구성하는 하나의 프레임을 뽑아서 이 이미지를 입력으로 하는 신경망을, 그리고 다른 한 쪽은 여러 프레임에서 뽑은 옵티컬 플로우를 누적(stacking)하여, 이것을 입력으로 하는 신경망을 사용하였다. 최종단에서는 앙상블(ensemble) 구조로 이 두개의 신경망에서 나온 결과를 종합하여 사람의 행동을 판별하도록 하였다.

[2, 3, 1]은 컨볼루션 신경망을 다양한 방법으로 구성하여 좋은 성능을 보여주었으나, 3차원 컨볼루션 신경망의 특성상, 많은 연산량과 파라미터가 필요하다는 문제점을 안고 있다. 또한 [8]의 경우는 2차원 컨볼루션 신경망을 사용하여, 연산량과 파라미터 수에서는 다른 연구에 비하여 장점이 있으나, [2]와 더불어 옵티컬 플로우와 같이 전처리 과정이 필요한 입력을 사용하여 이를 위한 추가적인 신경망이나 알고리즘을 통한 많은 연산이 필요하다는 단점이 존재한다. 또한 다양한 분야로의 응용 측면에서도 카메라 입력 이외에 추가적인 정보가 필요한 측면은 단점으로 작용할 수 있다.

[19]에서는 이러한 단점을 극복할 수 있는 신경망 구조를 새롭게 제안하였는데, 단순한 2차원 컨볼루션 신경망을 통하여 매 프레임마다 특성을 추출하고 이를 LSTM의 입력으로 넣어, 제스처를 판단하는 방법이다. 일반적으로 LSTM의 경우 연산량이 매우 많은 것으로 알려져 있는데, [19]에서는 이를 해결하기 위하여 사용하는 32비트 부동소수점 값을 갖는 일반적인 파라미터를 2비트로 줄여서 많은 메모리 사용량과 연산량 문제를 해결하였다. 하지만, 이 논문에서는 본 논문과 같이 Cambridge 데이터 세트를 이용하였으나 인식 실패율이 20%가 넘어 성능 면에서 부족하다는 단점이 있다.

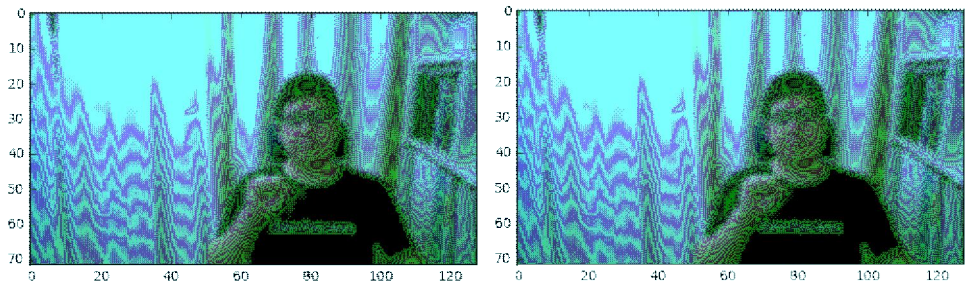
본 논문에서는 이러한 3차원 컨볼루션 신경망이 가진 기본적인 한계인 많은 연산량과 많은 수의 파라미터를 효과적으로 줄이면서, 정확도에서도 좋은 결과를 낼 수 있는 방법들에 대하여 소개하고 그 실험 결과에 대해서 분석하도록 하겠다.

제 3 장 효율적인 제스처 인식 방법

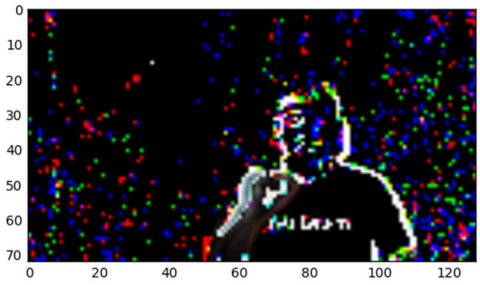
3.1 입력 영상으로 차영상을 이용하는 방법

2.1에서 살펴본 것과 같이 대부분의 컨볼루션 신경망에서 활성화 함수로 ReLU를 사용하고 있다. ReLU의 특징은 식 (4)에서 보는 것과 같이, 음수의 입력이 들어오는 경우 출력으로 0을 내보낸다는 것이다. 따라서 컨볼루션 신경망의 입력층을 제외한 모든 은닉층에서 입력으로 들어오는 특성맵의 많은 부분이 0의 값을 갖게 되며, [12]에 따르면 ImageNet 데이터 세트를 이용한 컨볼루션 신경망의 경우 약 44%의 값이 0을 갖는다고 한다. [12]에서는 이렇게 0이 많은 경우, 컨볼루션 연산을 적용할 때 모든 값에 대하여 다 곱셈 연산을 하는 것은 비효율적이기 때문에, 0의 값을 갖는 픽셀에서는 곱셈을 건너뛸 수 있는 새로운 하드웨어 구조를 제안하였다. 하지만, 이 구조를 사용한다고 하더라도 입력층의 곱셈 연산을 줄일 수는 없는데, 이는 입력 데이터는 제어가 불가능하기 때문이다. 하지만, 영상 데이터를 입력으로 사용하는 3차원 컨볼루션 신경망의 경우에는 인접한 프레임 간의 차를 입력으로 사용하면, 곱셈 연산을 줄일 수 있다. 일반적인 영상 데이터의 경우 [그림 3-1]에서 보는 것과 같이 인접한 프레임 간에는 움직임이 있는 일부를 제외하고는 큰 차이가 없다는 특징이 있다. 따라서 [그림 3-

2]와 같이 인접한 프레임 이미지 간에 차를 구하여 이를 새로운 영상 데이터로 만들고, 이것을 3차원 컨볼루션 신경망의 입력으로 사용하면, 입력 데이터의 크기도 한 프레임만큼 줄어들지만, 입력 데이터 자체에도 많은 양의 0을 포함하게 되어 그만큼 곱셈 연산의 수를 줄일 수가 있다. 또한, 영상 내에 존재하는 배경은 대부분 움직임이 없거나 매우 적기 때문에, 이렇게 차영상을 사용하면 배경을 어느정도 제거할 수 있는 효과가 있고, 움직임이 많은 곳의 값들이 커짐으로 인하여 원본 영상을 사용하는 것과 비교하였을 때 더 효과적으로 학습할 수 있다는 장점이 있다.



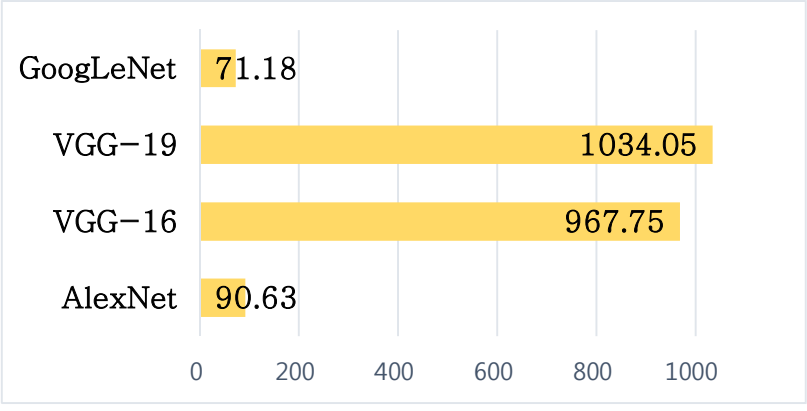
[그림 3-1] 연속된 프레임 영상



[그림 3-2] [그림 3-1]의 두 프레임 간 차영상

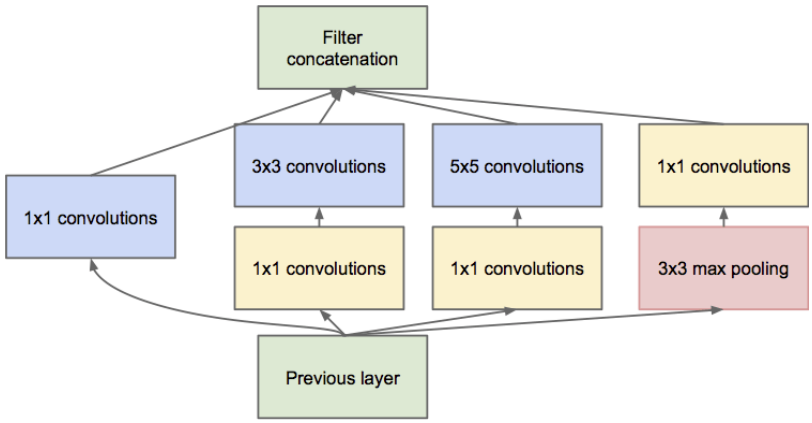
3.2 3차원 분해 인셉션 구조

ImageNet 분류 대회에서 2014년에 우승을 차지한 GoogLeNet[6]은 인셉션 구조를 이용하여, 22개 층의 깊은 신경망 구조에도 불구하고, 2012년에 우승을 차지한 AlexNet의 1/5에 해당하는 파라미터로 높은 분류 성능을 보여주었다. 일반적으로 신경망의 깊이가 깊어질수록 파라미터의 수가 증가하게 되며, 이에 따라서 연산량 또한 증가하게 된다. 또한, 데이터의 양이 적을 경우 과적합(overfitting)에 빠지게 될 가능성도 높아진다. 특히 컨볼루션 신경망의 연산의 대부분이 컨볼루션층에서 일어나므로, 컨볼루션층에서의 연산량을 줄이는 것은 매우 중요하다. **[그림 3-3]**에서는 대표적인 컨볼루션 신경망들의 층별 평균 연산량을 나타내고 있다.



[그림 3-3] 대표적 컨볼루션 신경망의 층별 평균 연산량(Mops/Layer)

GoogLeNet은 이러한 문제를 해결하기 위하여 인셉션 구조를 반복해서 사용하였는데, 인셉션 구조([그림 3-4])의 핵심은 다양한 필터를 병렬적으로 사용한다는 것과, 필터의 수 즉 채널의 수를 줄이기 위하여, 1x1 컨볼루션을 사용하였다는 것이다. 또한 인셉션 v3에서는 인셉션 구조 내부에 있는 5x5 필터를 더 작은 3x3 필터 2개로 분해하는 방법과, n x n 필터를 n x 1과 1 x n 필터로 분해하여 더 적은 양의 파라미터로 비슷한 성능을 낼 수 있는 방법을 제안하였다. 5x5 필터를 3x3 필터 2개로 분리할 경우 입력 채널이 1개라고 가정하였을 때, 파라미터의 수가 25개에서 18개로 줄어드는 효과가 있고, 이 차이는 채널이 많아질수록 더 커지게 된다. 또한, 이렇게 분해한 3x3 필터 2개를 다시 1x3과 3x1 필터로 분해할 경우 전체 파라미터 수가 25개에서 12개 즉, 절반 이하로 줄어드는 효과를 가질 수 있다.

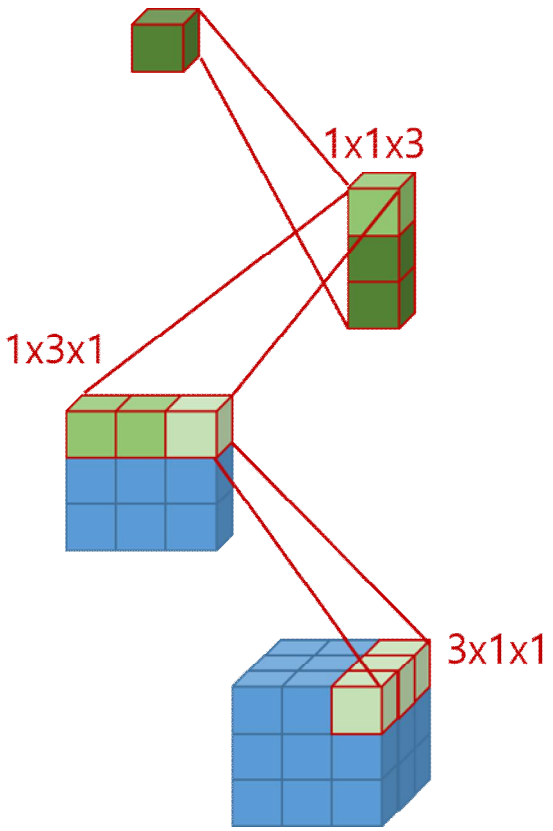


[그림 3-4] GoogLeNet의 인셉션 구조

앞에서 살펴본 바와 같이 3차원 컨볼루션 신경망의 경우 2차원 컨볼루션 신경망에 비하여 단순한 구조를 갖는다고 하더라도, 훨씬 더 많은 양의 파라미터 수와 연산량이 요구된다. 또한 위에서 살펴본 필터 분해 방법은 차원이 늘어남에 따라서 줄어드는 파라미터의 비율이 기하급수적으로 늘어나게 된다. 따라서, 이러한 인셉션 구조를 3차원으로 확장하여 3차원 컨볼루션 신경망에 적용하는 것은 제스처 인식에 있어서 효율성을 높일 수 있는 매우 좋은 방법이 될 수 있다.

[그림 3-6]은 일반적인 3차원 컨볼루션 신경망을 보여준다. 필터는 5x5x5(가로x세로x시간)을 사용하였으며, 각 계층 별로 필터의 개수는 각각 16, 32, 64개를 사용하였다. 활성화 함수로는 ReLU를 사용하였고, 각 컨볼루션층의 마지막에는 최대 풀링(max-pooling)이 적용되었다. **[그림 3-7]**는 일반적인 3차원 컨볼루션 신경망에 3차원 인셉션 구조를 적용한 신경망을 나타내고 있다. 기본적인 구조는 inception v1을 3차원으로 확장한 구조인데, 5x5x5 필터 대신 3x3x3 필터 2개를 사용하여 일부를 분해한 형태를 사용하였다. 마지막으로 **[그림 3-8]**에는 3차원 인셉션 구조에서 컨볼루션 필터를 최대한 분해하여 3x3x3 필터를 다시 3x1x1, 1x3x1, 1x1x3의 세 가지 필터로 치환한 구조를 보여주고 있다**[그림 3-5]**. 이렇게 필터를 분해할 경우, 5x5x5 필터 하나가 3x1x1 과 유사한 형태의 필터 6개로 분리되며, 채널을 1개로 가정하였을 때, 파라미터 수는 125개에서 18개로 줄어든다. 또한, 3x3x3 필터는 3x1x1 형태의 필터 3개로 분리되며, 역시 채널을 1개로 가정하였을 때, 파라미터 수는 27개에서 9개로 줄어들게 된다. **[그림**

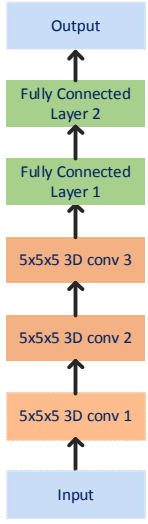
3-6]에 적용된 분해된 3차원 인셉션 구조를 사용한 신경망의 각 필터별 채널 수 및 cambridge 데이터 세트에 대한 출력 크기에 대한 자세한 정보는 **[표 3-1]**에 나타나 있다. 표에서 보는 것과 같이, **[그림 3-6]**에서 첫 번째 컨볼루션층에서 필요한 파라미터는 5x5x5x3x16 = 6,000 개가 필요하고, **[그림 3-8]**에 적용된 구조의 경우, 첫 번째 컨볼루션층에 필요한 파라미터 수는 606개로 약 1/10 수준이다. 또한 이 차이는 상위의 컨볼루션층으로 갈수록 더 커지게 된다.



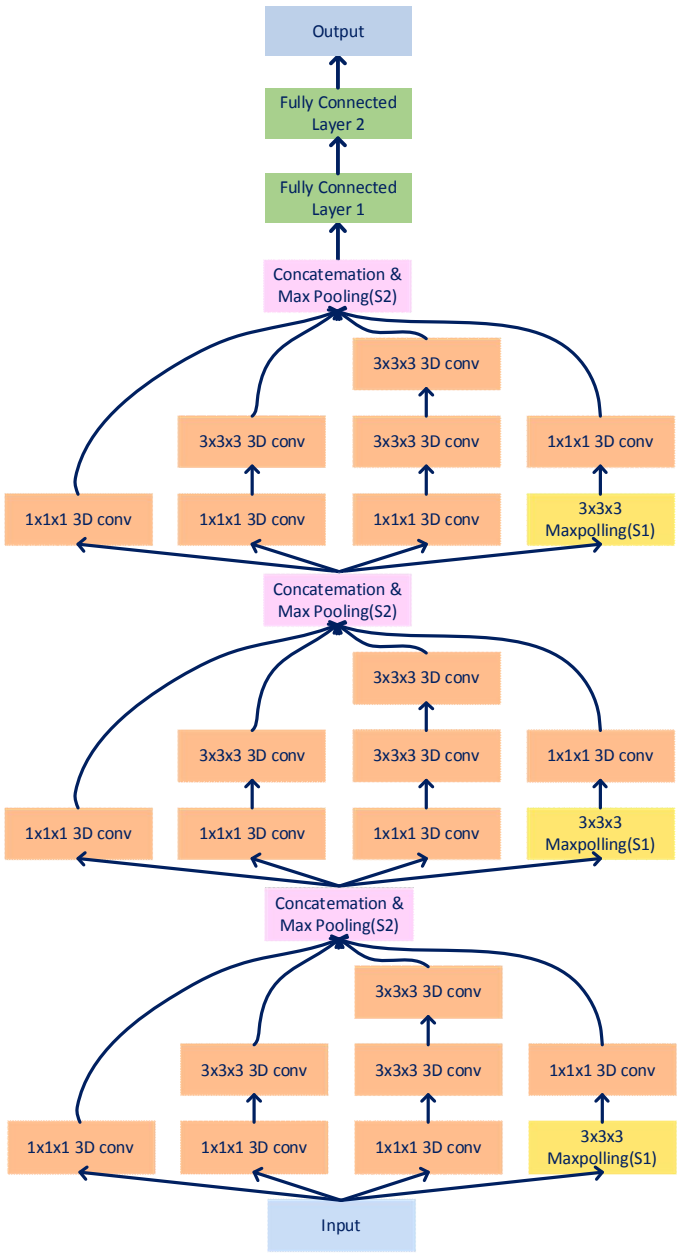
[그림 3-5] 3x3x3 컨볼루션 필터의 분해원리

[표 3-1] 3차원 분해 인셉션 구조의 출력 및 채널 구성

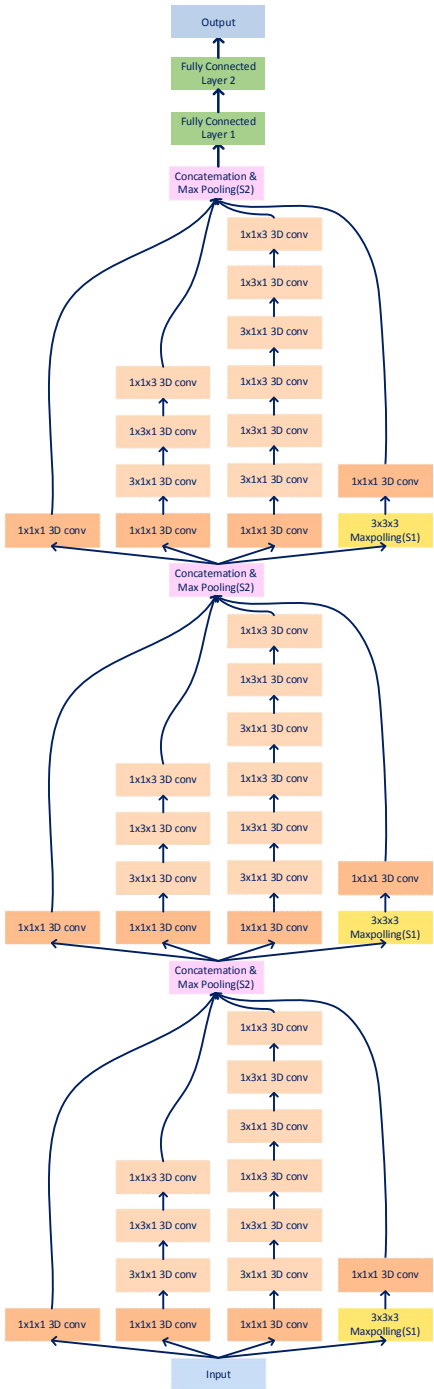
	Inception1	Inception2	Inception3
Output size (w x h x t x d)	32x32x32x16	16x16x16x32	8x8x8x64
#1x1	4	8	16
#3x3x3 reduce	6	12	24
#3x1x1	8	16	32
#1x3x1	8	16	32
#1x1x3	8	16	32
#5x5x5 reduce	2	4	6
#3x1x1	2	4	8
#1x3x1	2	4	8
#1x1x3	2	4	8
#3x1x1	2	4	8
#1x3x1	2	4	8
#1x1x3	2	4	8
Pool	2	4	8



[그림 3-6] 일반적인 3차원 컨볼루션 신경망



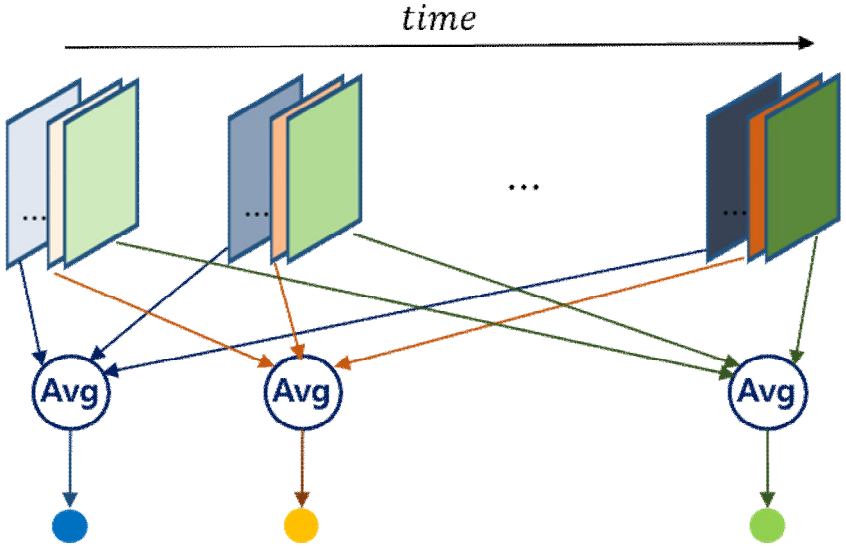
[그림 3-7] 인셉션 구조의 3차원 컨볼루션 신경망



[그림 3-8] 분해 인셉션 구조의 3차원 컨볼루션 신경망

3.3 3차원 글로벌 평균 풀링

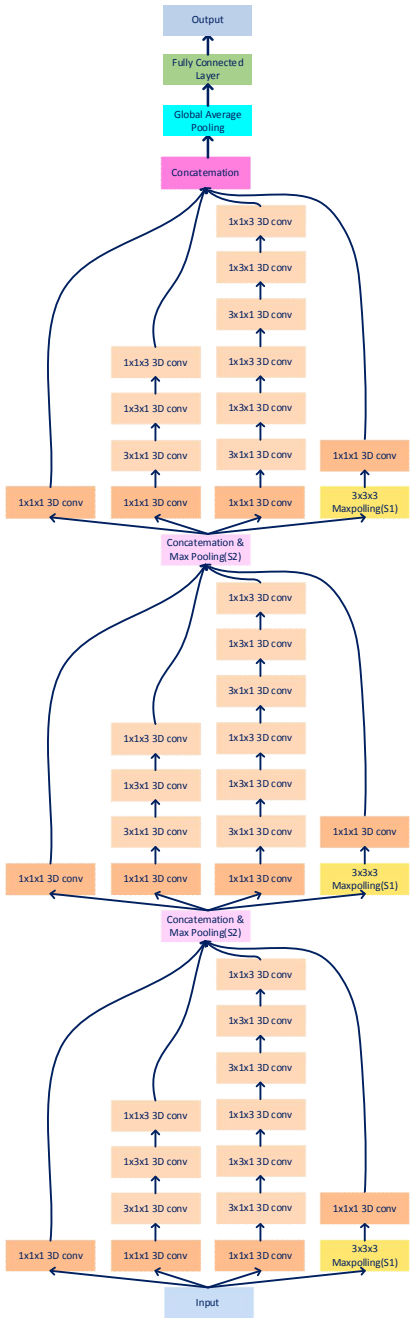
2.1에서 살펴본 것처럼, 컨볼루션 신경망에서는 맨 마지막 계층인 전결합층의 파라미터 수가 전체 파라미터의 대부분을 차지하게 된다. 또한 이러한 전결합층의 파라미터 수는 컨볼루션 신경망의 차원이 확장될수록 기하급수적으로 늘어나게 된다. 일반적으로 이렇게 많은 파라미터는 과적합 문제를 일으키며 이것은 일반화 성능에 악영향을 미치게 된다. 그래서 일반적으로 드랍아웃(dropout)과 같은 정규화(regulrarization) 방법을 많이 사용하는데, [13]에서는 이러한 전결합층의 많은 파라미터로 인한 과적합 문제 및 일반화 성능의 악화를 해결하기 위하여 전결합층 대신 글로벌 평균 풀링을 사용하는 방법을 제안하였다[그림 3-9].



[그림 3-9] 글로벌 평균 풀링

글로벌 평균 풀링은 [그림 3-9]에서 보는 것과 같이 식 (6)에서 살펴본 평균 풀링에서 H와 특성맵의 가로, 세로 길이가 같은 경우를 말한다. 글로벌 평균 풀링은 몇가지 장점을 갖는데, 그 첫째는 전결합층에 비하여 분류 문제를 푸는데 더 자연스럽다는 것이다. 아래쪽 컨볼루션층에서 각 계층별로 특성맵을 추출하고, 이 추출된 특성맵들을 바로 종합(평균값을 이용)하여 분류에 사용하기 때문에 학습이 진행될수록 각 특성들과 분류 카테고리 간의 관계를 강화하는 역할을 한다. 반면에 전결합층은 마지막 특성맵의 모든 픽셀값을 하나하나 일렬로 늘어놓고 그에 대한 파라미터를 학습해야 하므로 학습 과정에서 어떤 일이 일어날 지 예측하기 힘들다는 단점이 있다. 글로벌 평균 풀링의 두번째 장점은 파라미터 수가 0이라는 것이다. 앞서 언급한 것처럼 전결합층을 사용하면 매우 많은 파라미터로 인한 여러가지 문제들이 발생할 수 있는데, 글로벌 평균 풀링은 파라미터가 존재하지 않기 때문에 이런 문제가 발생하지 않는다. 세번째 장점은 글로벌 평균 풀링은 특성맵 한 장 전체의 평균값을 계산하는 방식이기 때문에 시간적 혹은 공간적 특성에 대한 변화에 강력하는 것이다. 그리고 마지막으로 일반적인 컨볼루션 신경망의 경우 항상 입력으로 받는 이미지나 영상의 크기가 고정되어야 하는데, 이는 전결합층에서 마지막 특성맵의 픽셀 값을 하나하나 벡터화하여 이용하기 때문이다. 글로벌 평균 풀링을 사용하게 되면 마지막 특성맵의 크기가 얼마가 되더라도 그 전체의 평균만 사용하므로 입력 이미지나 영상의 크기에 제약을 받지 않는다는 장점이 있다. GoogLeNet도 마지막 단에 이러한 글로벌 평균 풀링을

사용하였으나, 신경망의 확장성을 고려하여 글로벌 평균 풀링 이후에 작은 전결합층을 추가로 사용하였다. 본 논문에서도 [13]에서 제안한 글로벌 평균 풀링을 3차원으로 확장하고, 가장 마지막 단에 작은 크기의 전결합층을 사용하였다. 3차원 컨볼루션 신경망은 파라미터 수도 많을 뿐만 아니라, 움직임이 시간적으로도 공간적으로도 다양하게 분포할 수 있기 때문에 이러한 3차원 글로벌 평균 풀링을 사용하는 것이 파라미터 수도 획기적으로 줄일 수 있고, 시공간적인 변화에 잘 대응할 수 있는 방법이 된다. 이렇게 차영상을 입력으로 이용하는 인셉션 구조에 글로벌 풀링까지 적용한 3차원 컨볼루션 신경망을 본 논문에서 최종적으로 제안하며 [그림 3-10]에 제안하는 신경망의 구조를 나타내었다.



[그림 3-10] 글로벌 평균 풀링이 적용된 3차원 컨볼루션 신경망

제 4 장 실험 결과 및 분석

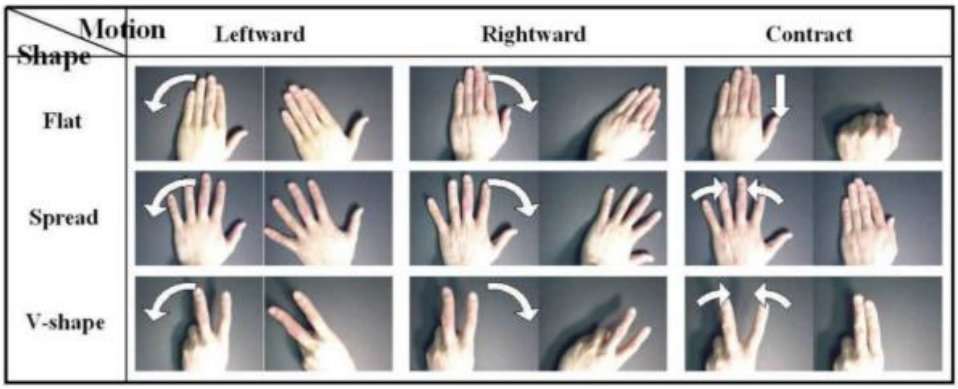
4.1 실험 환경

4.1.1 데이터 세트

본 논문에서는 실험을 위하여 아래와 같은 2가지 데이터 세트를 사용하였다.

4.1.1.1 Cambridge Hand Gesture 데이터 세트[10]

이 데이터 세트는 **[그림 4-1]**에서 보는 바와 같이 9개의 제스처로 구성되어 있다. 제스처는 3개의 손 형상(Flat, Spread, V-shape)과 3개의 모션(Leftward, Rightward, Contract)의 조합으로 이루어져 있으며, 모든 이미지의 해상도는 QVGA(320x240)이다. 그리고 각 제스처는 100개의 이미지 시퀀스 데이터로 구성되어 있는데 이것은 5개의 다른 조명과 10개의 임의의 모션과 2개의 피사체로 이루어진다. 각 이미지 시퀀스는 고정된 카메라에서 각각 촬영되었으며, 데이터 세트 내에는 각 제스처 별로 공간적 그리고 시간적으로 다양한 변화가 존재한다.



[그림 4-1] Cambridge Hand Gesture 데이터 세트

4.1.1.2 CAPP 데이터 세트

이 데이터 세트는 연구실에서 자체적으로 제작한 데이터 세트로 [그림 4-2]와 같이 구성되어 있으며, 스마트 TV에서 많이 사용하는 제스처들로 이루어져 있다. 각 제스처를 하나씩 분리하면, 총 20개의 제스처가 존재하며, 역시 각 제스처에 대하여 100개의 이미지 시퀀스로 구성되어 있다. 모든 이미지의 해상도는 720p(1280x720)이며, 30fps로 촬영되었다. 각 제스처마다 손의 공간적 변화를 다양하게 제작하였고, 손의 움직임 속도 또한 다양하게 촬영되었다.

[그림 4-2] CAPP 데이터 세트의 제스처 구성

4.1.2 학습 및 실험 조건

실험에 사용된 데이터 세트는 80%를 학습, 10%를 검증, 나머지 10%를 테스트에 사용하였다. 각 이미지 해상도는 정확도를 떨어뜨리지 않는 선에서 최대한 축소하였으며, 잘라내기(cropping)을 이용한 데이터 확장 기법을 사용하였다. 또한, 일반적인 3차원 컨볼루션 신경망에는 고정된 사이즈의 이미지 시퀀스 즉 고정된 해상도와 함께 고정된 프레임 수만큼의 데이터가 필요한 반면에 제스처가 수행되는 시간은 각 데이터에 따라서 차이가 있다. 따라서 이미지 리사이징(resizing) 이외에 제스처가 수행되는 시간을 고려하여 시간상으로 정규화하는 작업을 진행하였다. 즉, 컨볼루션 신경망에 32 프레임의 데이터가 입력으로 들어가는데, 제스처가 64 프레임에 걸쳐서 수행된다면, 약 2프레임마다 1장씩의 이미지를 모아서 입력으로 사용하였다. 입력 프레임 수와 제스처 수행 프레임 수가 정확하게 나누어지지 않는 경우, 보통 제스처가 시간 상으로 중간에 오는 경우가 많기 때문에 이러한 경우 시간상 중앙 쪽에서 더 촘촘히 샘플링 하는 방법을 이용하였다. 마지막으로 이렇게 생성된 이미지 시퀀스에 3장에서 제안한 차영상 사용을 위하여 인접한 프레임 간 차이를 이용한 차영상 시퀀스를 제작하고 최종적으로 이를 실험에 사용하였다.

최초 학습률(learning rate)은 0.001로 시작하여 학습률 감소 기법(learning rate decay)을 사용하였으며, 학습을 위하여 Adam optimizer를 사용하였다. 또한 조기 멈춤(early stopping) 기법을 사용하여, 과적합 현상이 일어나지 않도록 하였다.

학습과 테스트는 각 데이터 세트에 대하여 총 8개로 나누어 진행되었는데, 각각은 다음과 같다.

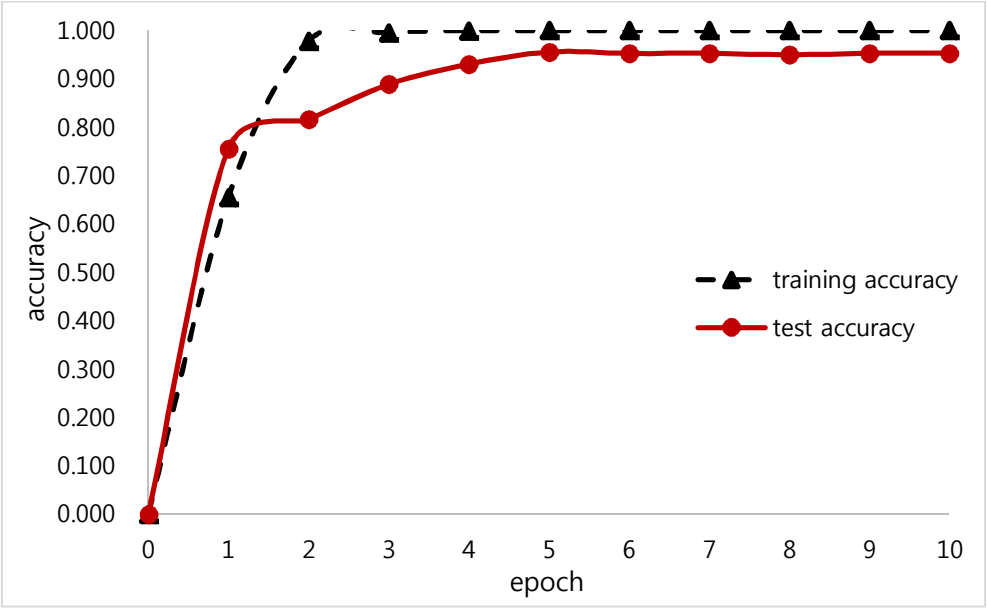
A. **일반적 3차원 컨볼루션 신경망 + 전결합층 + 원본 영상 사용**
B. **일반적 3차원 컨볼루션 신경망 + 전결합층 + 차영상 사용**
C. **일반적 3차원 컨볼루션 신경망 + 글로벌 평균 풀링 + 원본 영상 사용**
D. **일반적 3차원 컨볼루션 신경망 + 글로벌 평균 풀링 + 차영상 사용**
E. **3차원 인셉션 구조 컨볼루션 신경망 + 전결합층 + 차영상 사용**
F. **3차원 인셉션 구조 컨볼루션 신경망 + 글로벌 평균 풀링 + 차영상 사용**
G. **3차원 분해 인셉션 구조 컨볼루션 신경망 + 전결합층 + 차영상 사용**
H. **3차원 분해 인셉션 구조 컨볼루션 신경망 + 글로벌 평균 풀링 + 차영상 사용**

4.2 학습 및 테스트 결과

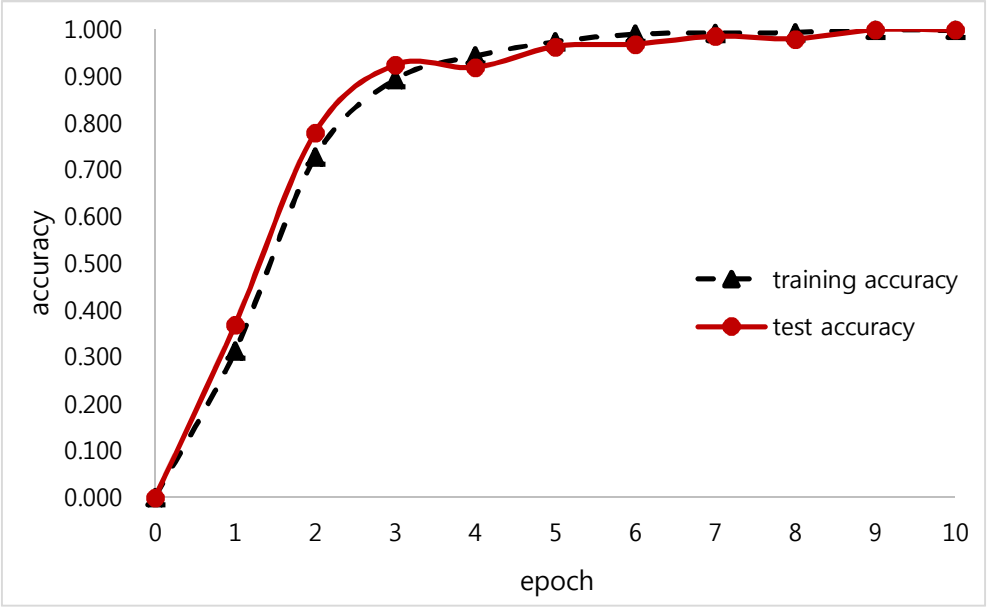
Cambridge 데이터 세트를 이용한 실험 결과를 **[그림 4-3]~[그림 4-10]**에 나타내었고, CAPP 데이터 세트를 이용한 실험 결과는 **[그림 4-11]~[그림 4-18]**에 나타내었다. 그리고 각 데이터 세트에 대한 테스트 결과를 **[그림 4-19], [그림 4-20]**에, 최종 테스트 정확도를

[표 4-1]에 나타내었다.

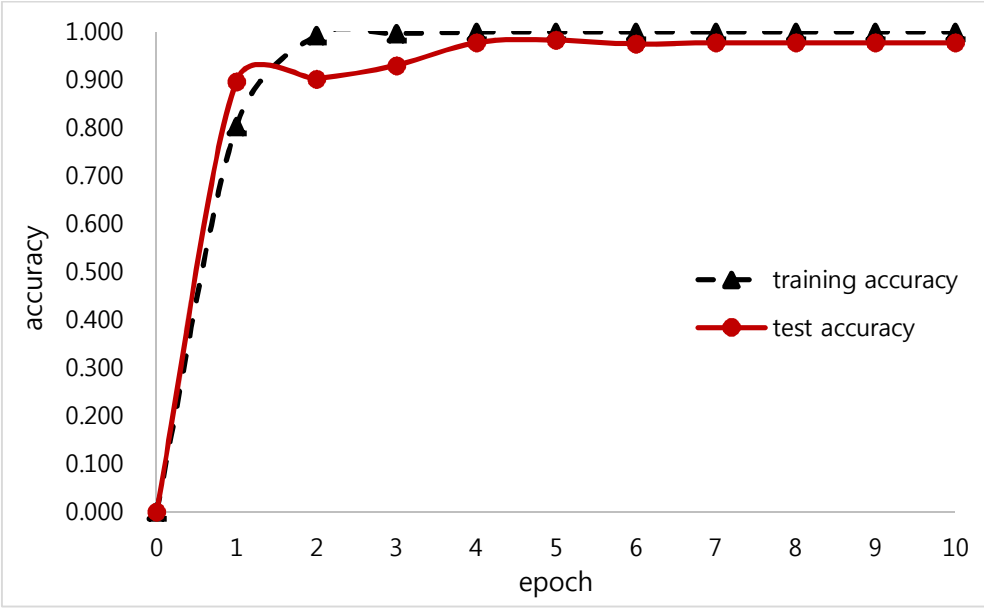
Cambridge 데이터 세트의 경우 인셉션 구조와 글로벌 평균 풀링을 사용할 경우 99.1%의 테스트 정확도를 보였으며, CAPP 데이터 세트에서는 98.5%의 테스트 정확도를 보였다. 차영상을 사용할 경우에는 원본 영상을 사용하는 것보다 정확도가 떨어지는 경우가 일부 있었으나, 학습 속도는 더 빠른 것을 확인할 수 있으며, 정확도도 글로벌 평균 풀링을 사용하면 원본 영상과 같은 수준으로 회복되는 것을 볼 수 있다. 일반적 구조에 비하여 인셉션 구조를 사용하는 경우에도 대부분 정확도가 향상되는 것을 볼 수 있는데, 이것은 인셉션 구조가 적은 파라미터를 가지고 있음에도 불구하고 다양한 필터를 통하여 더욱 다양한 특성맵을 뽑아낼 수 있기 때문으로 분석할 수 있다. 또한, 3장에서 예측한 것과 같이 글로벌 평균 풀링을 사용하는 것이 전결합층을 사용하는 것에 비하여 테스트 정확도가 상당부분 향상되는 결과를 볼 수 있는데, 이는 글로벌 평균 풀링을 사용하는 것이 다양한 시공간 상의 변화에 강한 신경망 구조임을 증명해주는 것이라 할 수 있다.



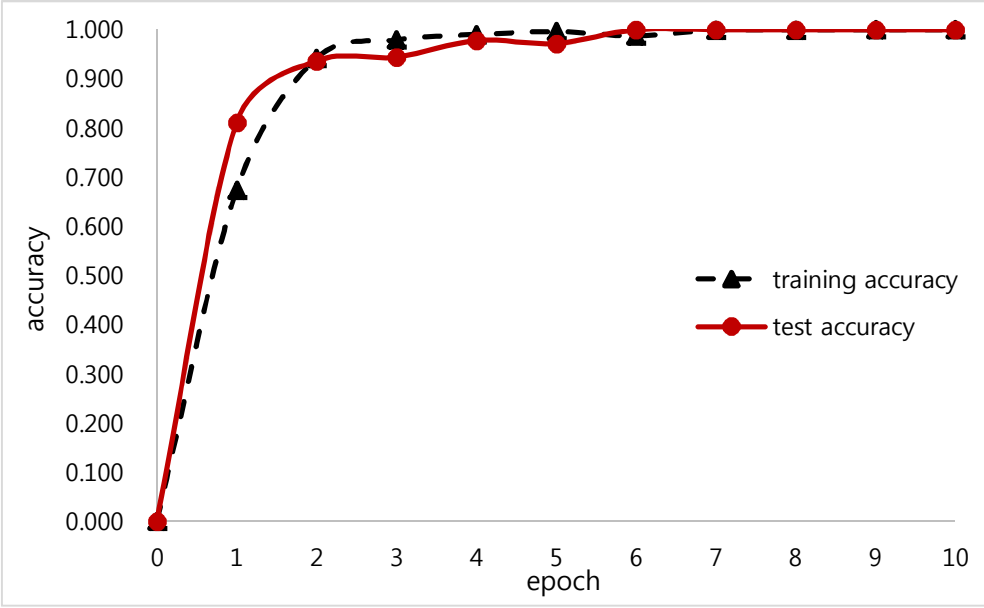
[그림 4-3] Cambridge / 일반 3D-CNN / 전결합층 / 원본 영상



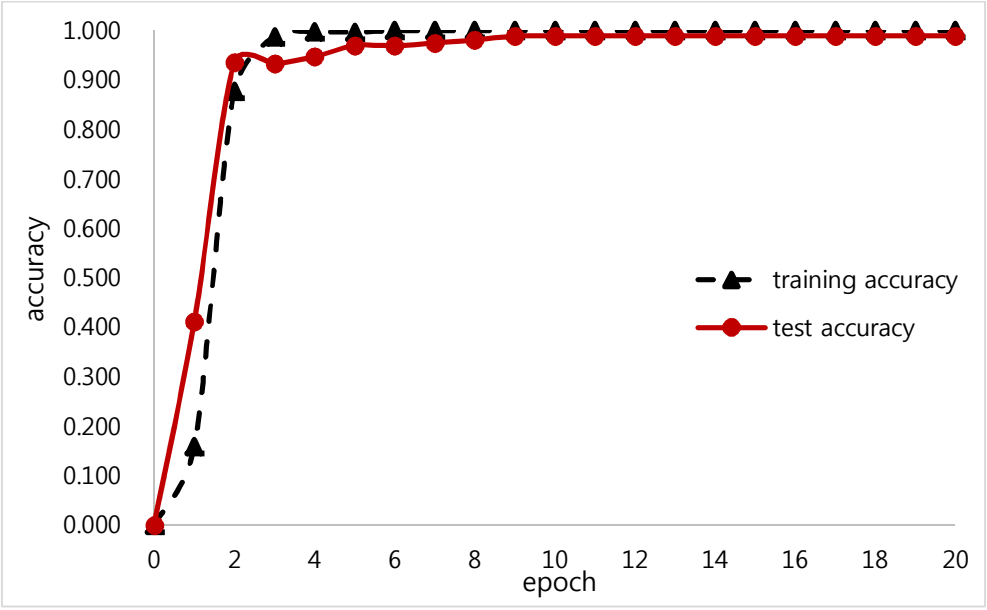
[그림 4-4] Cambridge / 일반 3D-CNN / 전결합층 / 차영상



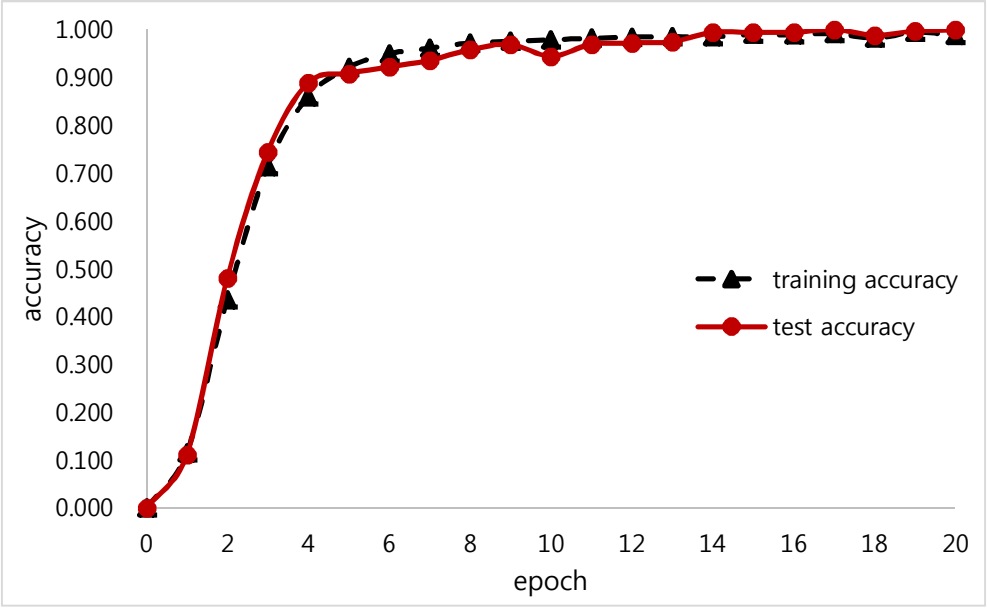
[그림 4-5] Cambridge / 일반 3D-CNN / 평균 풀링 / 원본 영상



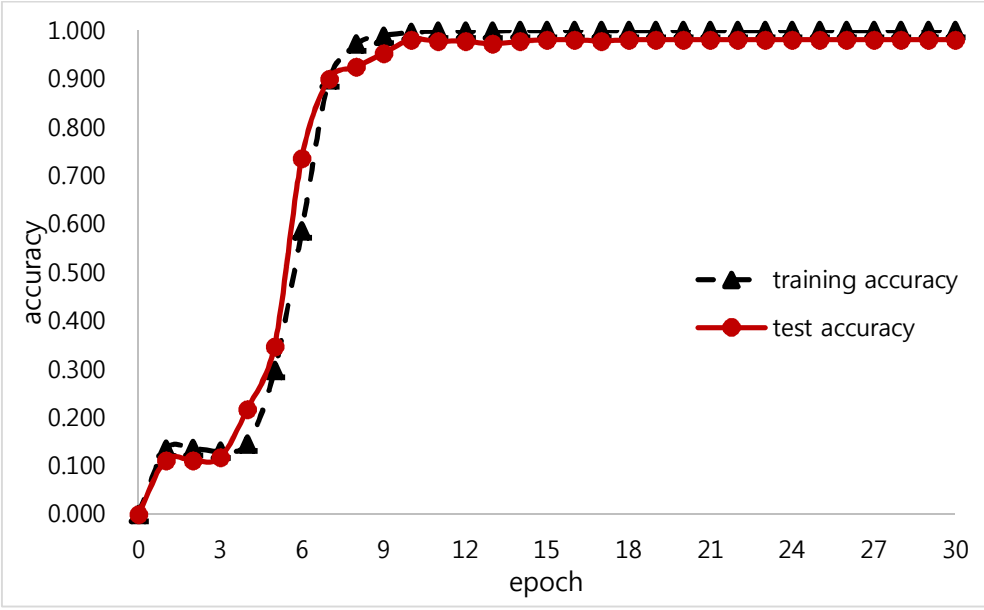
[그림 4-6] Cambridge / 일반 3D-CNN / 평균 풀링 / 차영상



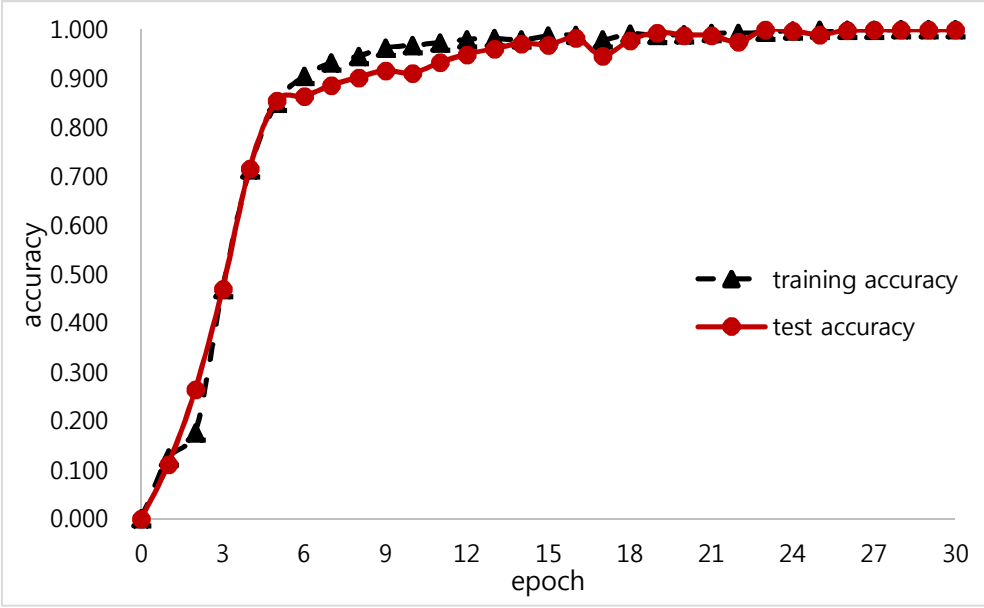
[그림 4-7] Cambridge / 인셉션 구조 / 전결합층 / 차영상



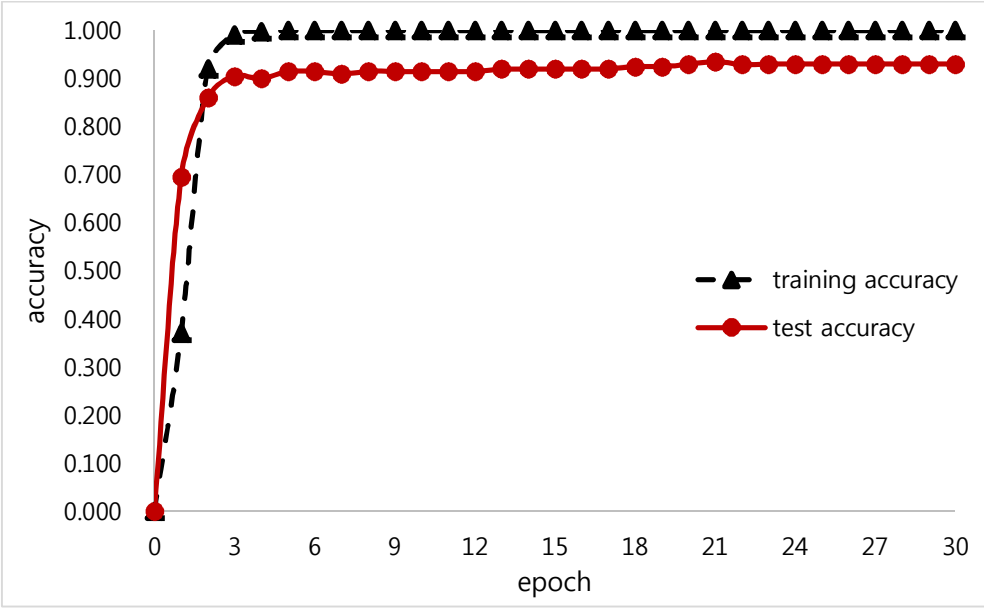
[그림 4-8] Cambridge / 인셉션 구조 / 평균 풀링 / 차영상



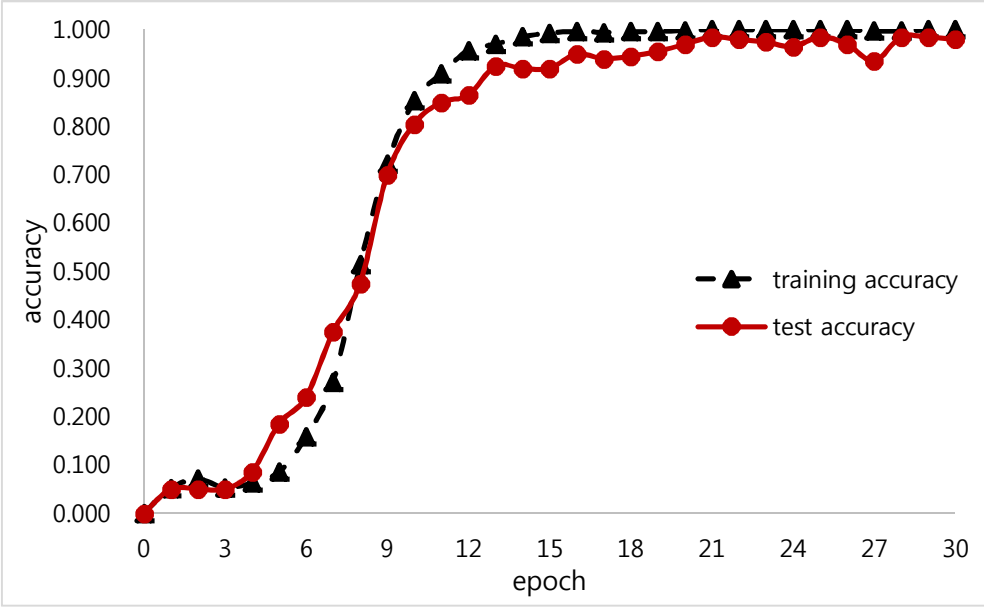
[그림 4-9] Cambridge / 분해 인셉션 구조 / 전결합층 / 차영상



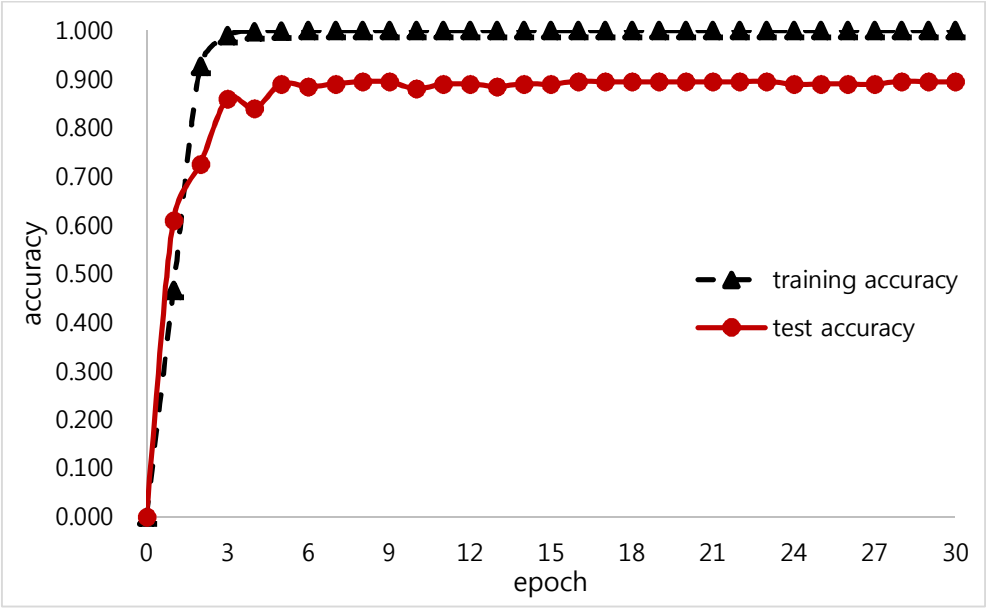
[그림 4-10] Cambridge / 분해 인셉션 구조 / 평균 풀링 / 차영상



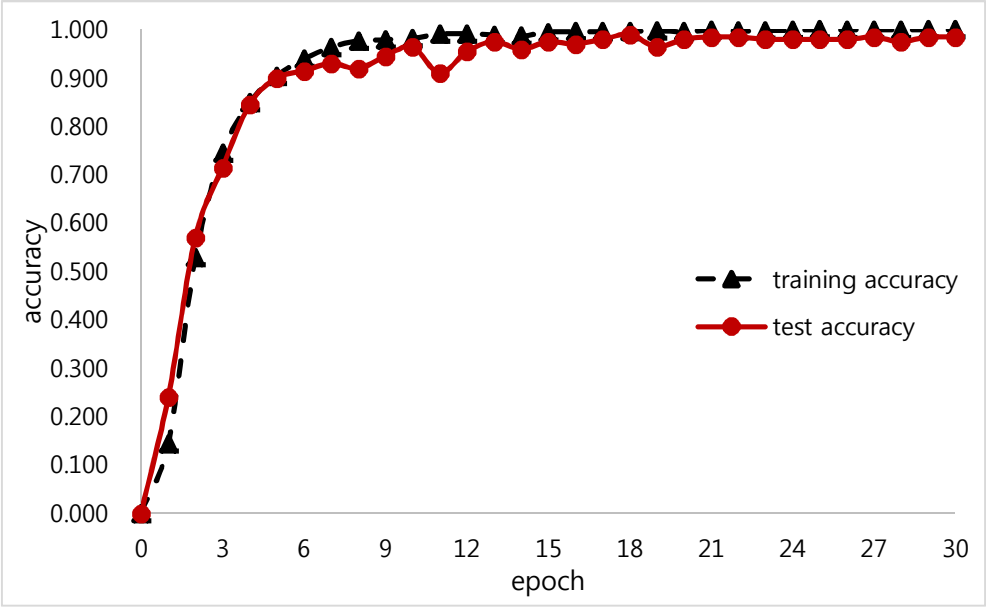
[그림 4-11] CAPP / 일반 3D-CNN / 전결합층 / 원본 영상



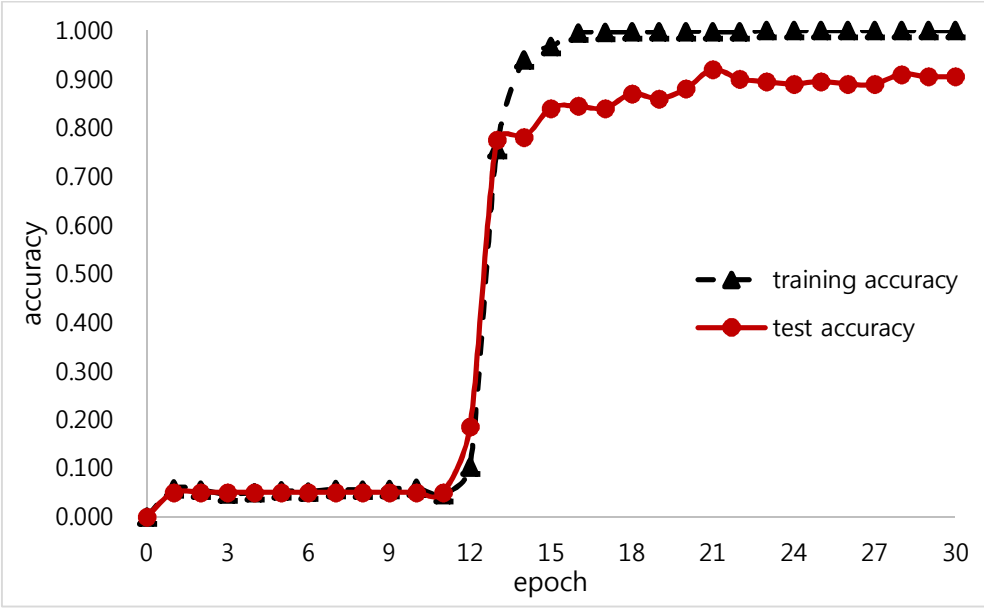
[그림 4-12] CAPP / 일반 3D-CNN / 전결합층 / 차영상



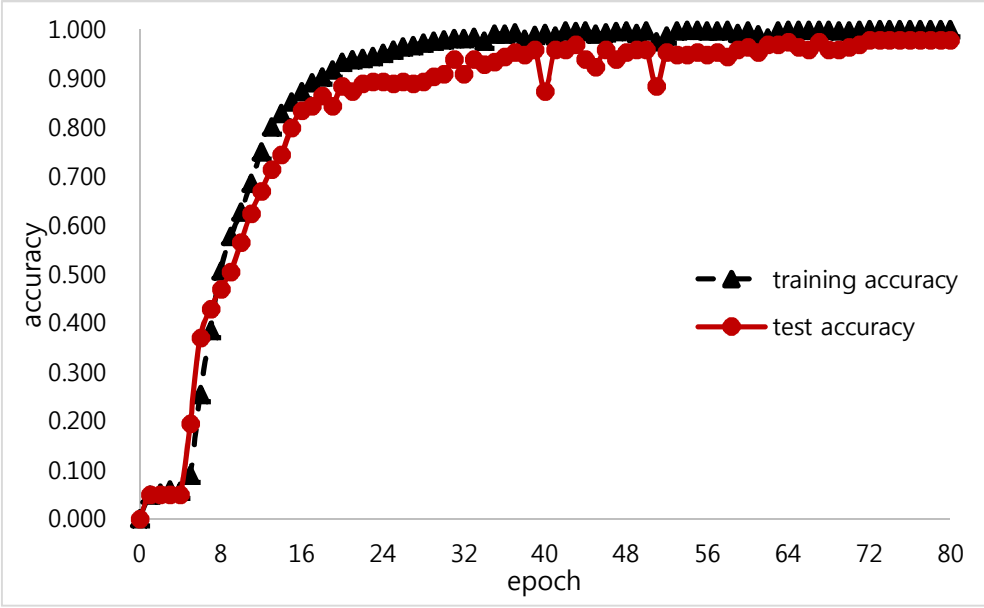
[그림 4-13] CAPP / 일반 3D-CNN / 평균 풀링 / 원본 영상



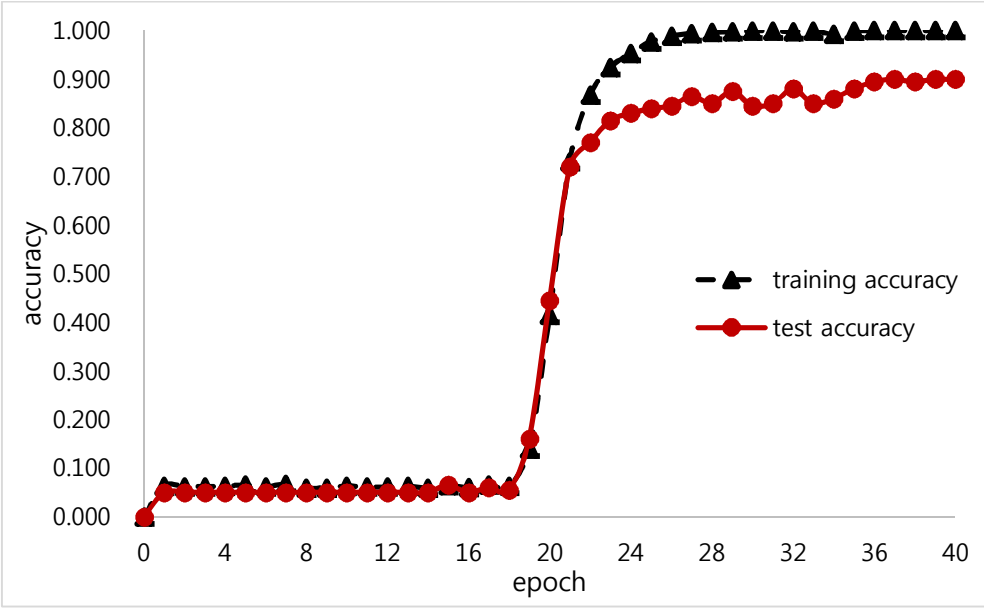
[그림 4-14] CAPP / 일반 3D-CNN / 평균 풀링 / 차영상



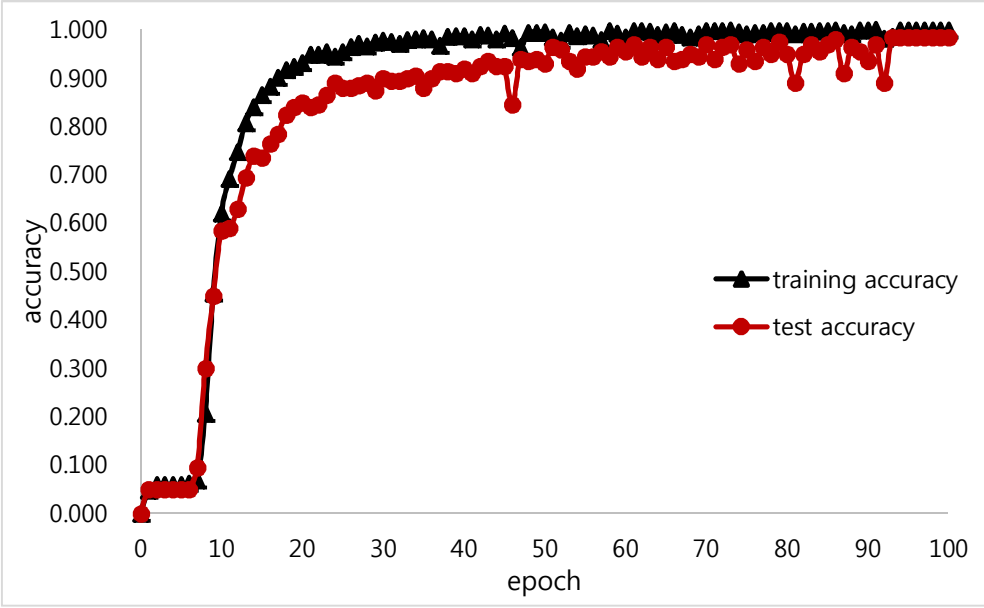
[그림 4-15] CAPP / 인셉션 구조 / 전결합층 / 차영상



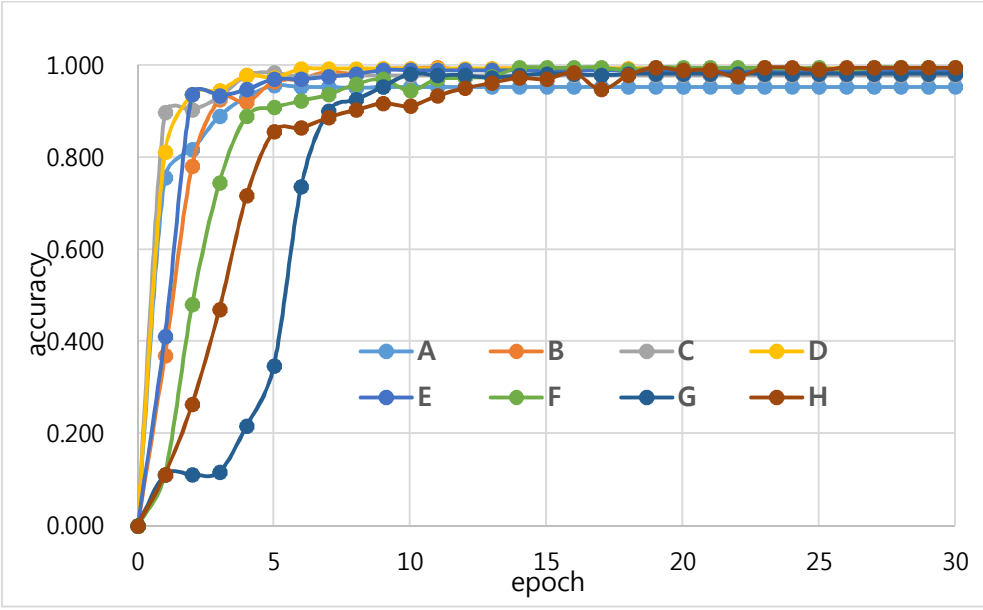
[그림 4-16] CAPP / 인셉션 구조 / 평균 풀링 / 차영상



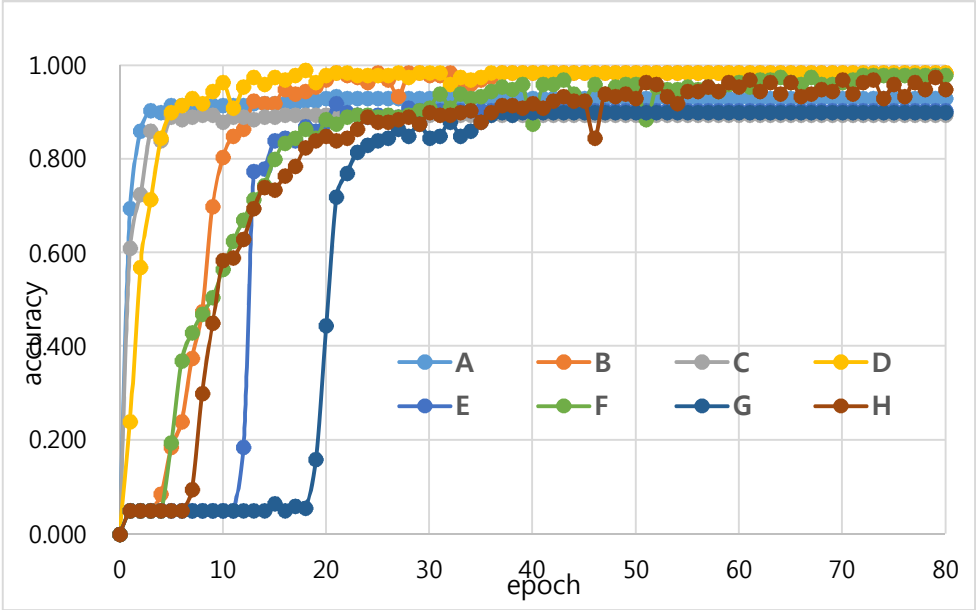
[그림 4-17] CAPP / 분해 인셉션 구조 / 전결합층 / 차영상



[그림 4-18] CAPP / 분해 인셉션 구조 / 평균 풀링 / 차영상



[그림 4-19] Cambridge 데이터 세트 테스트 결과



[그림 4-20] CAPP 데이터 세트 테스트 결과

[표 4-1] 각 데이터 세트 별 테스트 정확도

Test accuracy		Cambridge dataset	CAPP dataset
A	nor+fc+org	0.953	0.930
B	nor+gav+org	0.989	0.985
C	nor+fc+diff	0.978	0.895
D	nor+gav+diff	0.989	0.985
E	inc+fc+diiff	0.989	0.905
F	inc+gav+diff	0.991	0.980
G	fac+fc+diff	0.981	0.905
H	fac+gav+diff	0.991	0.985

4.3 신경망 구조별 파라미터 수 분석

[표 4-2]에 실험에 사용한 각 신경망 구조별로 필요한 파라미터의 수를 계산하여 나타내었다.

표에서 보는 것과 같이, 글로벌 평균 풀링과 함께 분해된 인셉션 구조를 사용한 3차원 컨볼루션 신경망의 경우, 일반적인 3차원 컨볼루션 신경망에 전결합층을 사용한 구조에 비하여 약 0.36%의 파라미터만을 가지고 학습에 성공하였음을 알 수 있다. 또한 파라미터 수는 인셉션 구조의 사용 유무 보다는 글로벌 평균 풀링의 사용 유무에 크게 좌우됨을 알 수 있다.

[표 4-2] 3차원 컨볼루션 신경망 구조별 파라미터 수

# of parameters	normal 3D CNN		3D CNN + inception		3D CNN + factorized inception	
	FC	GAP	FC	GAP	FC	GAP
1st conv layer	6,000		1,554		606	
2nd conv layer	64,000		6,496		2,632	
3rd conv layer	128,000		26,088		11,280	
1st fc layer	4,194,304	1,280	4,194,304	1,280	4,194,304	1,280
2nd fc layer	5,120		5,120		5,120	
sum	4,397,424	199,280	4,233,562	35,418	4,213,942	15,798
ratio	100.000%	4.532%	96.274%	0.805%	95.828%	0.359%
Memory usage(KB)	17,589.70	797.12	16,934.25	141.67	16,855.77	63.19

또한 현재 임베디드 시스템에 많이 사용되고 있는 ARM의 cortex-A 시리즈에는 L2 cache 사이즈가 최소 128KB에서 512KB로 구성되어 있는데(최대 사이즈는 1MB에서 8MB로 구성), 본 논문에서 제안하는 신경망의 경우 전체 파라미터를 저장하기 위한 메모리 요구량이 약 63KB이기 때문에, 프리패치와 같인 추가적인 연산 없이도 ARM core를 사용하는 다양한 임베디드 시스템에, 학습된 파라미터를 모두 cache에 올려놓고 추론(inference)에 사용하는 것이 가능하다고 판단할 수 있다.

4.4 신경망 구조별 곱셈 연산량 분석

[표 4-3]에 각 신경망 구조 별로 추론에 필요한, 곱셈 연산의 수를 분석하여 표시하였다.

[표 4-3] 3차원 컨볼루션 신경망 구조별 곱셈 연산 수

# of multiplications	normal 3D CNN		3D CNN + inception		3D CNN + factorized inception	
	FC	GAP	FC	GAP	FC	GAP
1st conv layer	786,432,000		203,685,888		58,982,400	
2nd conv layer	2,097,152,000		212,860,928		93,323,264	
3rd conv layer	1,048,576,000		104,398,848		47,251,456	
1st fc layer	8,388,608	1,280	8,388,608	1,280	8,388,608	1,280
2nd fc layer	5,120		5,120		5,120	
sum	3,940,553,728	3,932,161,280	529,339,392	520,946,944	207,950,848	199,558,400
ratio	100.000%	99.787%	13.433%	13.220%	5.277%	5.064%
Cambridge Dataset	3,717,207,040	3,708,814,592	471,492,600	666,786,040	191,199,846	182,807,398
ratio	94.332%	94.119%	11.965%	16.921%	4.852%	4.639%
CAPP Dataset	3,495,433,216	3,487,040,768	414,053,179	405,660,731	174,566,810	166,174,362
ratio	88.704%	88.491%	10.507%	10.295%	4.430%	4.217%

표에서 보는 것과 같이 본 논문에서 제안하는 구조의 3차원 평균 풀링과 분해된 인셉션 구조를 사용하는 컨볼루션 신경망의 경우 일반적인 3차원 컨볼루션 신경망에 비하여 약 10%의 곱셈 연산 만으로 동일한 학습 결과를 나타낼 수 있음을 알 수 있다. 파라미터 수와는 반대로 곱셈 연산의 경우 컨볼루션층에서 많이 일어나기 때문에, 글로벌 평균 풀링의 사용 여부 보다는 인셉션 구조의 사용 여부에 따라서 크게 영향을 받는 것을 볼 수 있다.

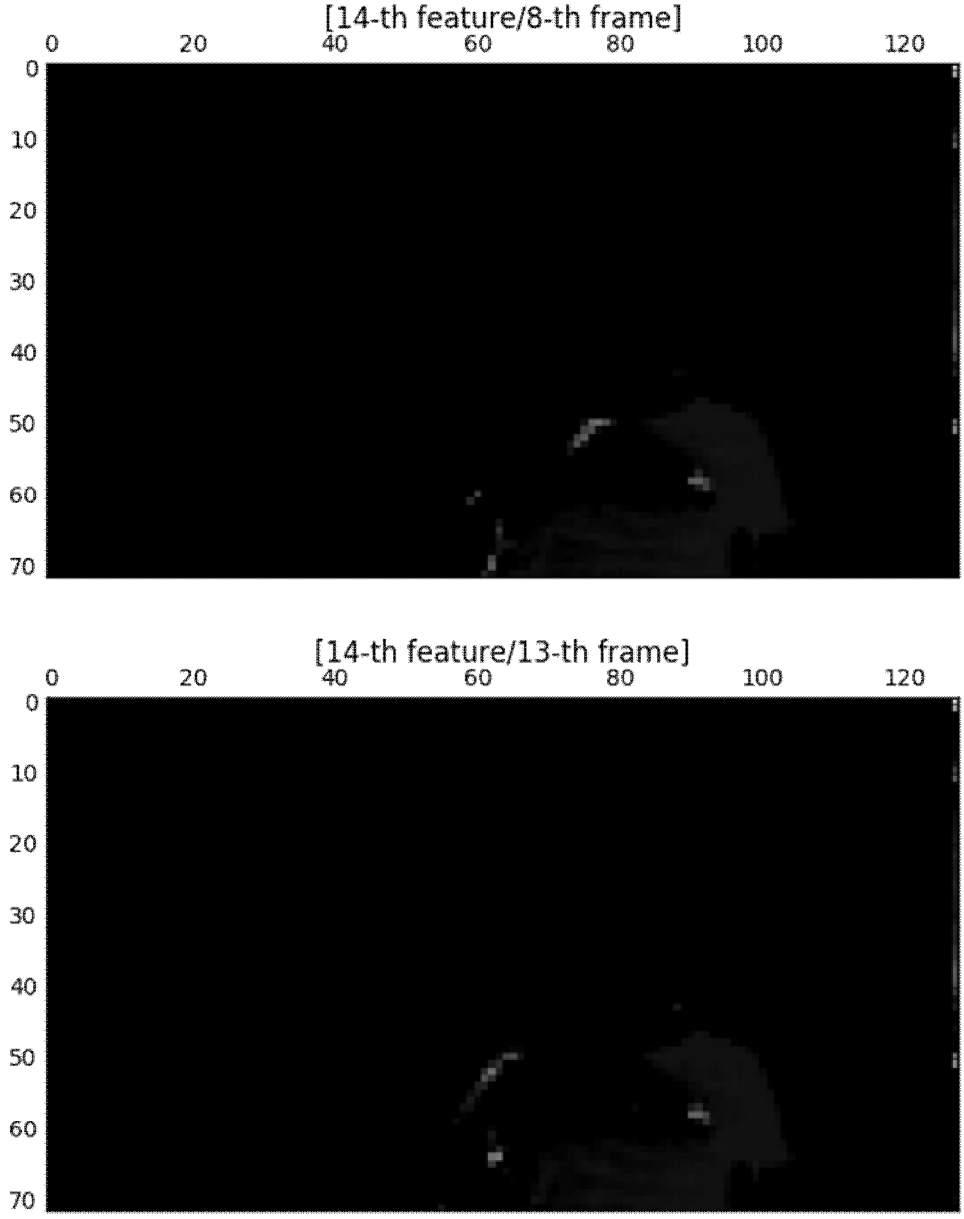
또한 표 아래쪽에는, 각 데이터 세트별로 차영상을 이용함으로 인하여 얻을 수 있는 연산량 감소를 계산하였다. 각 데이터 세트마다 차영상을 생성하였을 경우에 발생하는 0의 비율이 다르기 때문에 그 결과에 약간 차이가 있지만, 1~2% 정도의 추가 연산량 감소를 얻을 수 있음을 표를 통해 확인할 수 있다.

4.5 차영상을 이용한 효과 분석

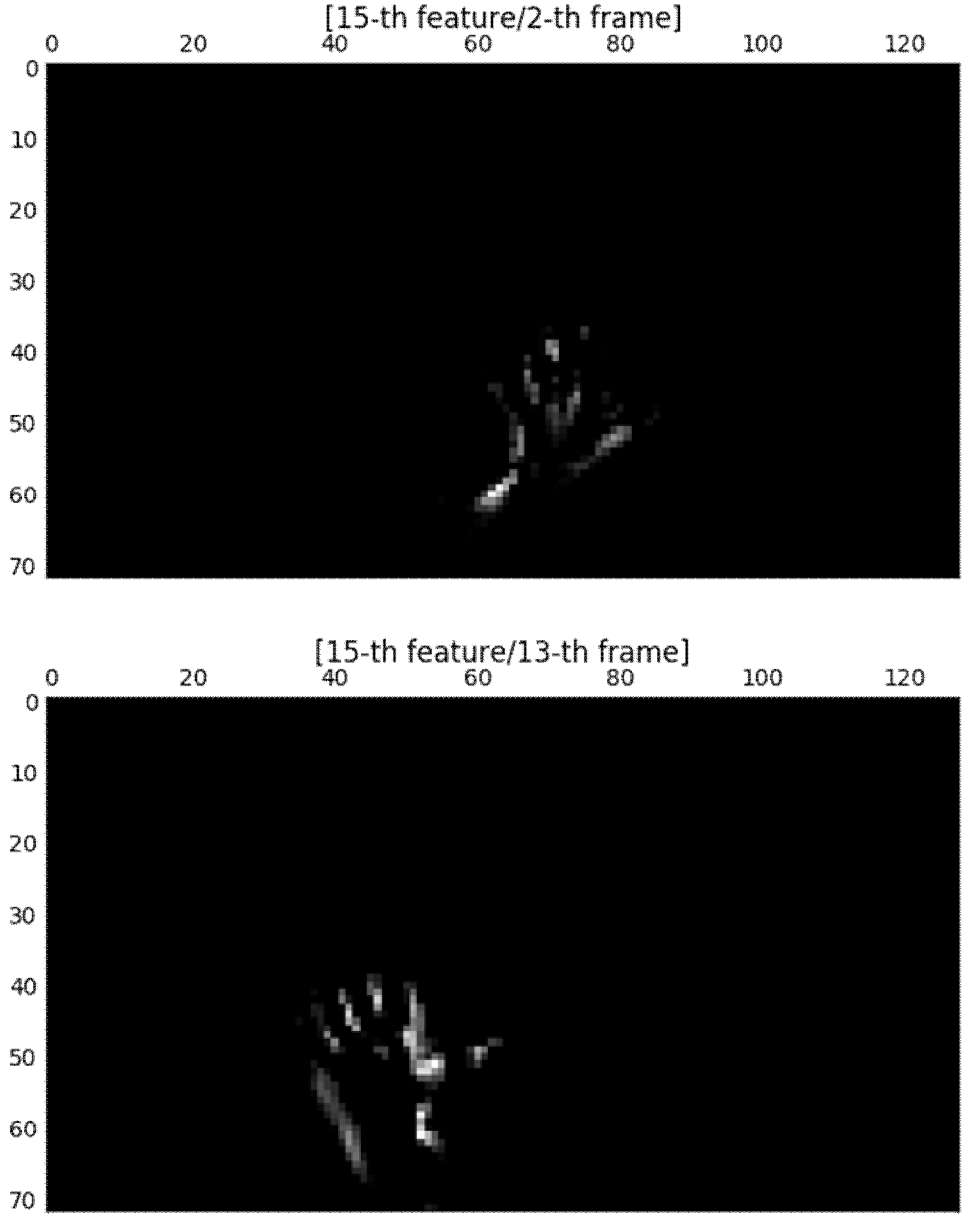
실험에 사용한 2가지 데이터 세트에 대하여 인접한 프레임 간의 차영상을 이용할 경우 이미지 내에 0이 얼마나 많이 발생하는 지를 조사해보았다. 그 결과, Cambridge 데이터 세트에서는 약 28.4%의 픽셀값이 0이었고, CAPP 데이터 세트에서는 약 61.9%의 픽셀값이 0이었다. Cambridge 데이터 세트의 경우, 이미지 내에서 손이 차지하는 비중이 크고, 프레임 간의 조명 변화가 심하여 CAPP 데이터 세트에 비하여 차영상에서 0의 비율이 작은 것을 확인할 수 있었다. 원본 영상에 대해서도 같은 방법으로 계산하였는데, Cambridge 데이터 세트의 원본 영상에는 0값을 갖는 픽셀이 존재하지 않았고, CAPP 데이터 세트에서는 그 비율이 약 5.3%였다. 이렇게 차영상을 입력으로 이용하면, 차영상 계산을 위한 추가적인 뺄셈 연산이 필요하지만 입력 이미지 시퀀스에서 0의 값의 비율이 커짐으로 인하여 곱셈 연산의 수를 줄일 수 있으므로, 전체적인 연산비용 측면에서 이득을 볼 수 있다.

또한 차영상을 이용할 경우, 움직임이 없는 배경과 같은 영역의 상당부분을 제거할 수 있기 때문에, 학습에 불필요한 정보를 제거해주는 효과를 기대할 수 있다. 본 논문에서는 이를 확인해보기 위하여, 학습이 끝난 이후의 3차원 컨볼루션 신경망에서 원본 영상과 차영상에 따라서 컨볼루션층에서 나온 출력, 즉 특성맵을 시각화하여 어떤 차이가 있는지 살펴보았다. **[그림 4-21]**에는 CAPP 데이터 세트의 원본 영상을 입력으로 이용한 경우의 첫 번째 컨볼루션층의 출력을, **[그림 4-**

22]에는 CAPP 데이터 세트의 원본 차영상을 입력으로 이용한 첫 번째 컨볼루션층의 출력을 일부 나타낸 것이다. 원본 영상의 활성맵의 경우는 사람의 손 이외의 신체부위 즉 얼굴이나 몸 부분들이 특성맵으로 뽑혀져 나온 경우가 많고, 차영상의 경우에는 움직임이 있는 손과 관련된 특성맵이 많이 나타나는 것을 확인할 수 있었다.



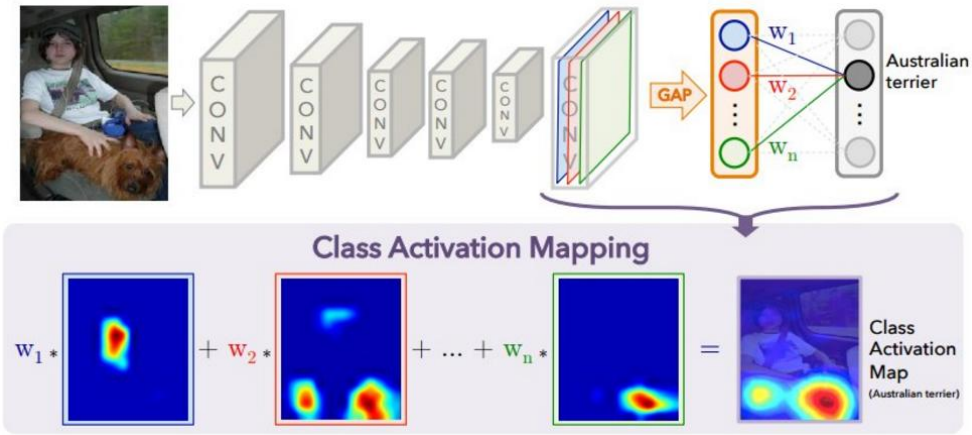
[그림 4-21] 원본 영상을 사용한 경우의 첫번째 컨볼루션층 출력 특성맵



[그림 4-22] 차영상을 상용한 경우의 첫번째 컨볼루션층 출력 특성맵

4.6 클래스 활성화 맵을 이용한 학습 분석(Weakly Supervised Learning)

클래스 활성화 맵(Class activation map)은 컨볼루션 신경망이 특정 클래스를 식별하기 위하여 특성맵의 어떤 영역에 집중하여 학습하였는지를 나타내는 방법 중 하나이다.



[그림 4-23] 클래스 활성화 맵(출처 : [16])

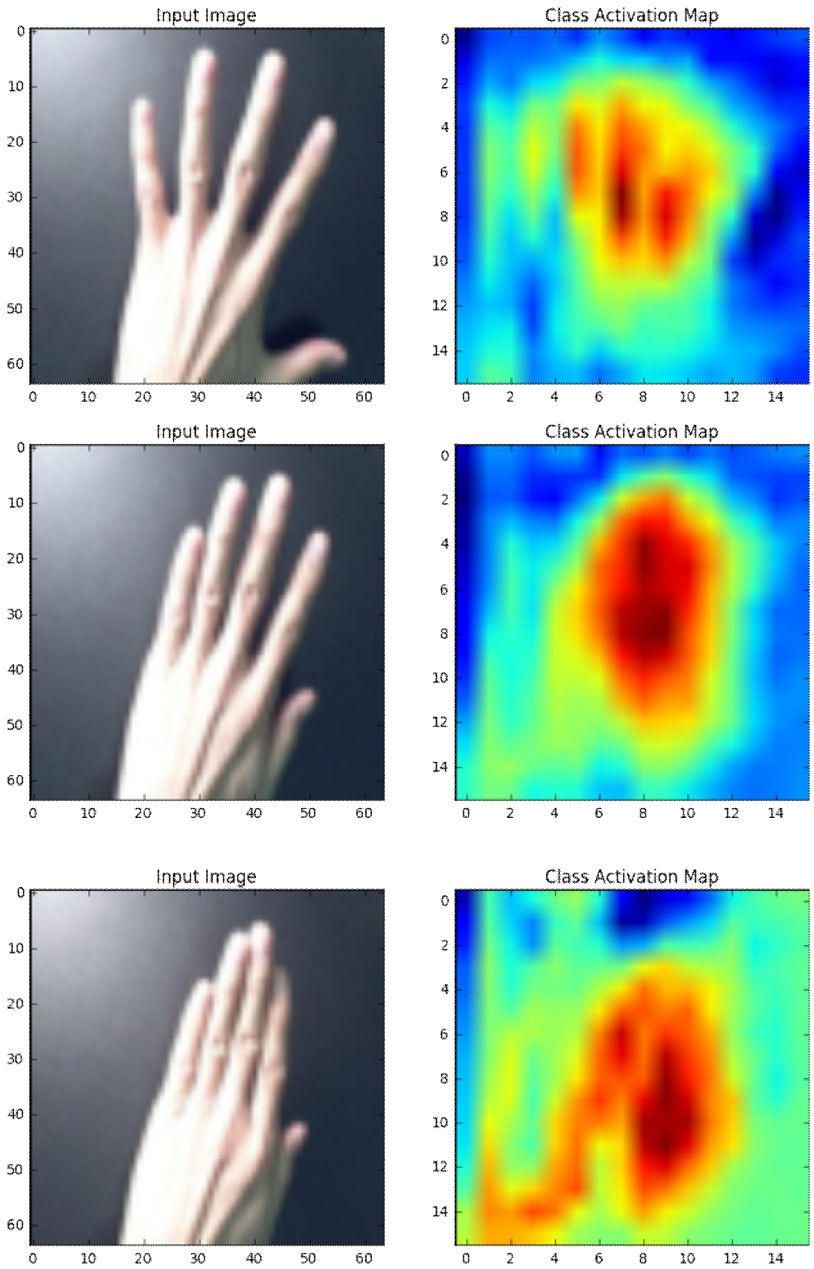
[그림 4-23]과 같이 글로벌 평균 풀링을 사용한 컨볼루션 신경망에서 어떤 클래스를 식별했다는 것은, 글로벌 평균 풀링에 이어지는 전결합층의 파라미터(그림에서 w1, w2와 같은) 값이 마지막 특성맵의 평균에 가중치로 작용하여 그 합의 값을 이용한 것이다. 따라서 마지막 특성맵에 이 파라미터를 곱하여 가중치 합(weighted sum)을 구하면, 그림에서 보는 것과 같은 클래스 활성화 맵을 얻을 수

있다. 이를 이용하면 학습된 신경망이 어떤 부분이 집중하여 학습을 진행 하였는지를 시각화 할 수 있는데, 본 논문에서 제안한 3차원 컨볼루션 신경망도 글로벌 평균 풀링을 사용하였기 때문에, 이에 맞게 클래스 활성 맵을 변형하면 비슷한 과정을 통하여 각 프레임 별로 클래스 활성화 맵을 구할 수 있다. 이러한 과정을 통하여 얻은 활성화 맵을 각 데이터 세트별로 아래 그림에 나타내었다. **[그림 4-24]**에 Cambridge 데이터 세트의 원본 영상을 이용한 입력 시퀀스와 클래스 활성 맵을, **[그림 4-25]**에 Cambridge 데이터 세트의 차영상을 입력으로 사용한 이미지 시퀀스와 클래스 활성 맵을 처음과 마지막 프레임을 포함하여 일부 프레임에 대하여 나타내었다. 또한, **[그림 4-26]**은 원본 영상을 사용한 CAPP 데이터 세트의 이미지 시퀀스와 클래스 활성 맵을, 마지막으로 **[그림 4-27]**은 차영상을 사용한 CAPP 데이터 세트의 이미지 시퀀스와 클래스 활성 맵을 나타내고 있다. 이 그림들에 나타난 모든 데이터 세트에서 손이나 그 주변 영역의 활성도가 매우 높게 나타나는 것을 볼 때, 학습이 제대로 이루어졌음을 확인할 수 있다.

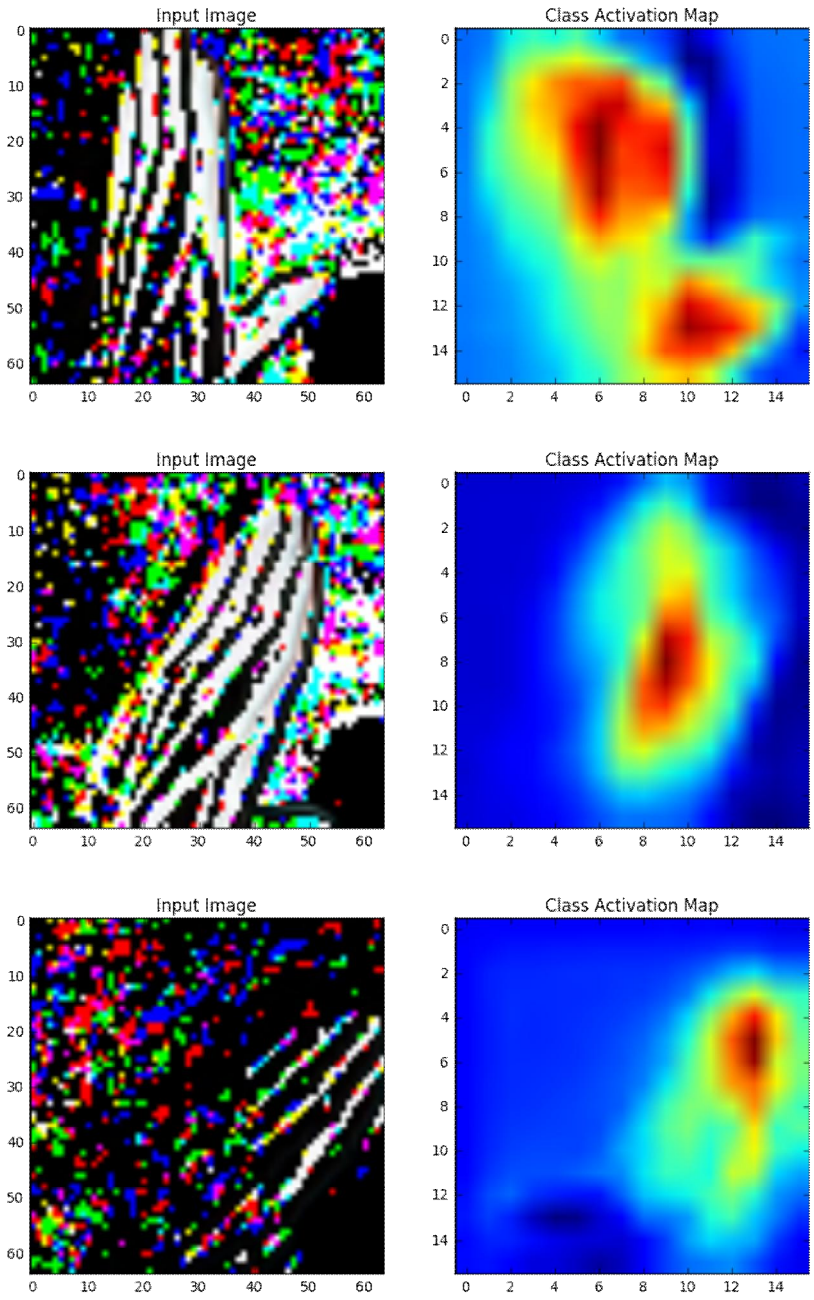
Cambridge 데이터 세트의 결과를 좀 더 자세히 분석해 보면, 데이터의 특성상 이미지 대부분을 손이 차지하고 있기 때문에, 손 전체보다는 손가락과 같이 제스처의 특징이 될 수 있는 부분에 활성도가 높게 나타나는 것을 볼 수 있다. 또한 이미지 시퀀스의 첫 프레임과 마지막 프레임의 경우 전체적으로 다른 프레임에 비하여 밝은 활성화 맵을 갖는데, 이 것은 첫 프레임과 마지막 프레임을 제스처를

구분하는데 다른 프레임보다 중요하게 고려되었다는 의미를 갖는다. Cambridge 데이터 세트의 특성상 중간 움직임 없이 맨 첫 동작과 마지막 동작만을 가지고도 9개의 제스처를 구분이 가능하기 때문에 이것은 매우 자연스러운 결과이다. 또한 **[그림 4-28]**을 보면 특이한 결과를 볼 수 있는데, flat contract 제스처의 경우 손가락이나 손 부분에 활성도가 높게 나타나는 것이 아니라, 첫 프레임에서 손가락 윗부분이 위치하던 배경 쪽에 활성도가 매우 높게 나타나는 것을 볼 수 있다. 이것은 Cambridge 데이터 세트를 구성하는 9개의 제스처 중에 해당 제스처 하나만 손가락을 굽히는 동작이 들어가기 때문에, 손가락 윗부분의 정보 만으로도 제스처를 구분해 낼 수 있기 때문으로 예상할 수 있다.

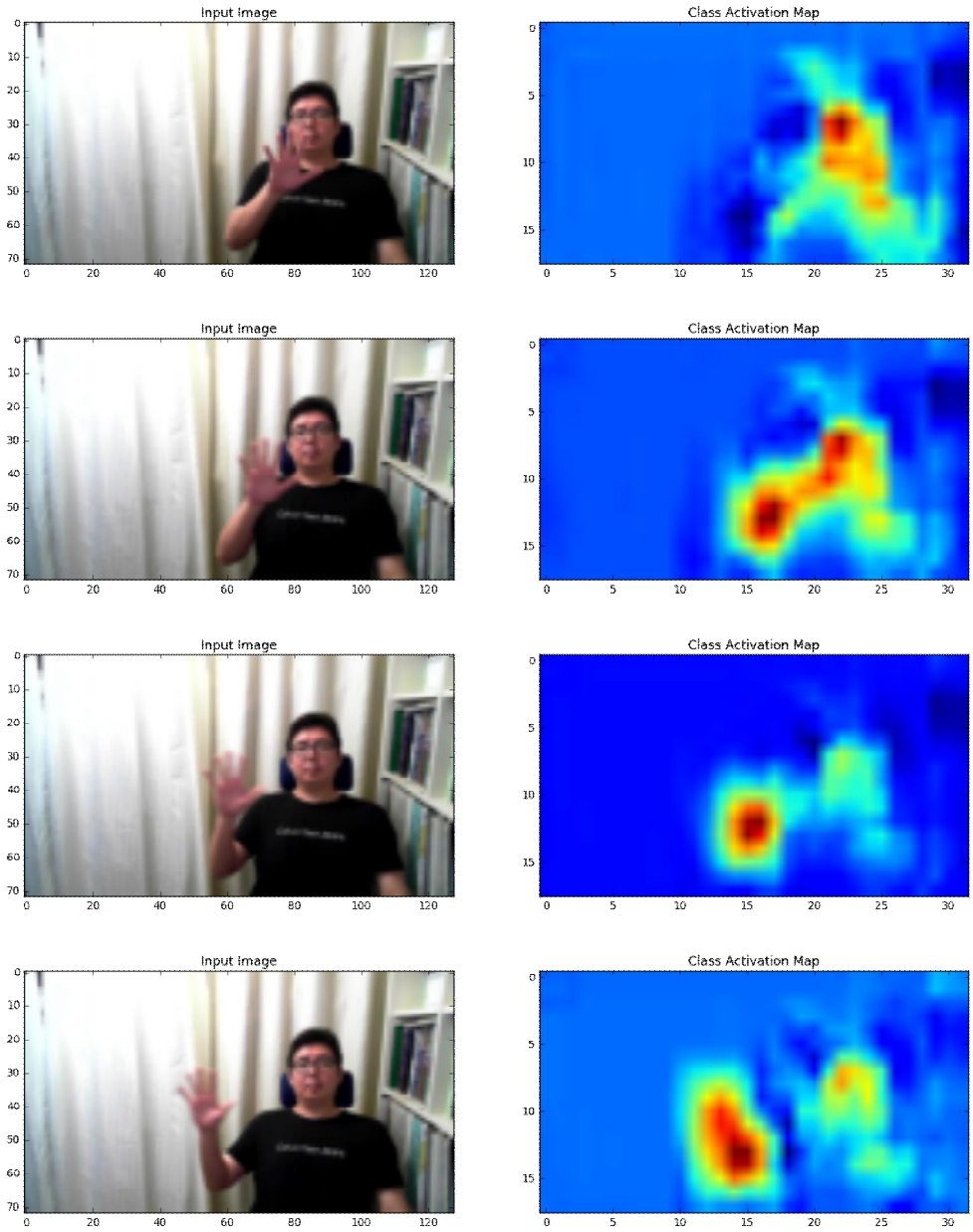
CAPP 데이터 세트의 경우 Cambridge 데이터 세트에 비하여 전체 이미지 영역에서 손이 차지하는 부분이 작은 것이 특징이고, 이로 인하여 클래스 활성화 맵 역시 손가락 보다는 손 전체를 포함하는 영역 특히 움직임이 있는 영역 쪽으로 집중되는 것을 볼 수 있다. 다만 원본 영상을 사용하는 경우와 차영상을 사용하는 경우를 잘 살펴보면 차영상에서는 활성화 맵이 손에만 집중되지만, 원본 영상에서는 얼굴과 같은 손을 제외한 나머지 주변 영역에도 활성도가 나타나는 것을 볼 수 있다. 제스처를 인식하는 데 있어서 손 외에 다른 얼굴이나 사람의 몸과 같은 부분은 꼭 필요한 정보가 아니므로 차영상을 사용하였을 경우에 원본 영상에 비하여 불필요한 정보가 많이 사라지게 되고, 이로 인해 학습이 빨라지는 것을 실험 결과로 확인할 수 있었다.



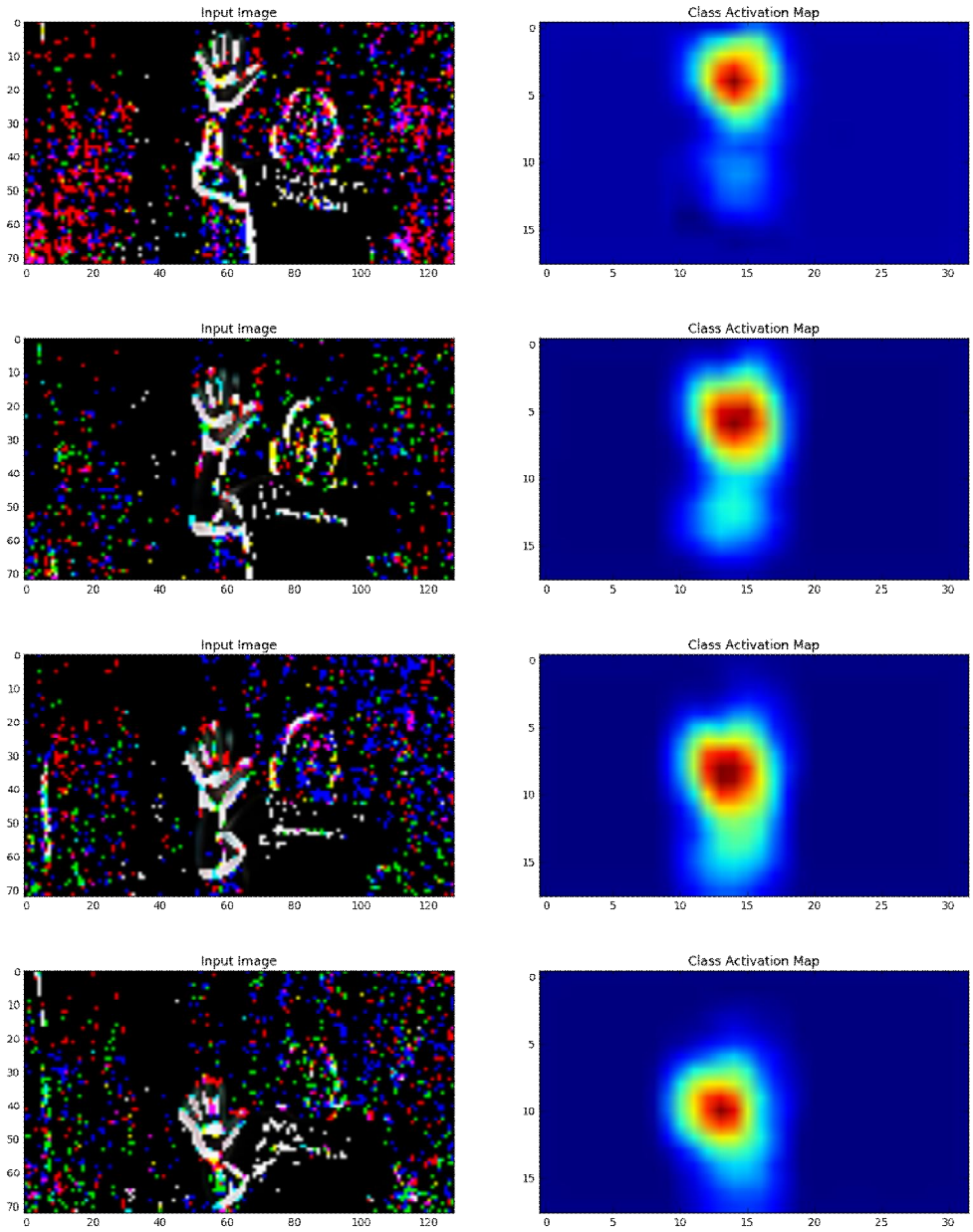
[그림 4-24] Cambridge 데이터 세트의 원본영상을 이용한 활성화 맵



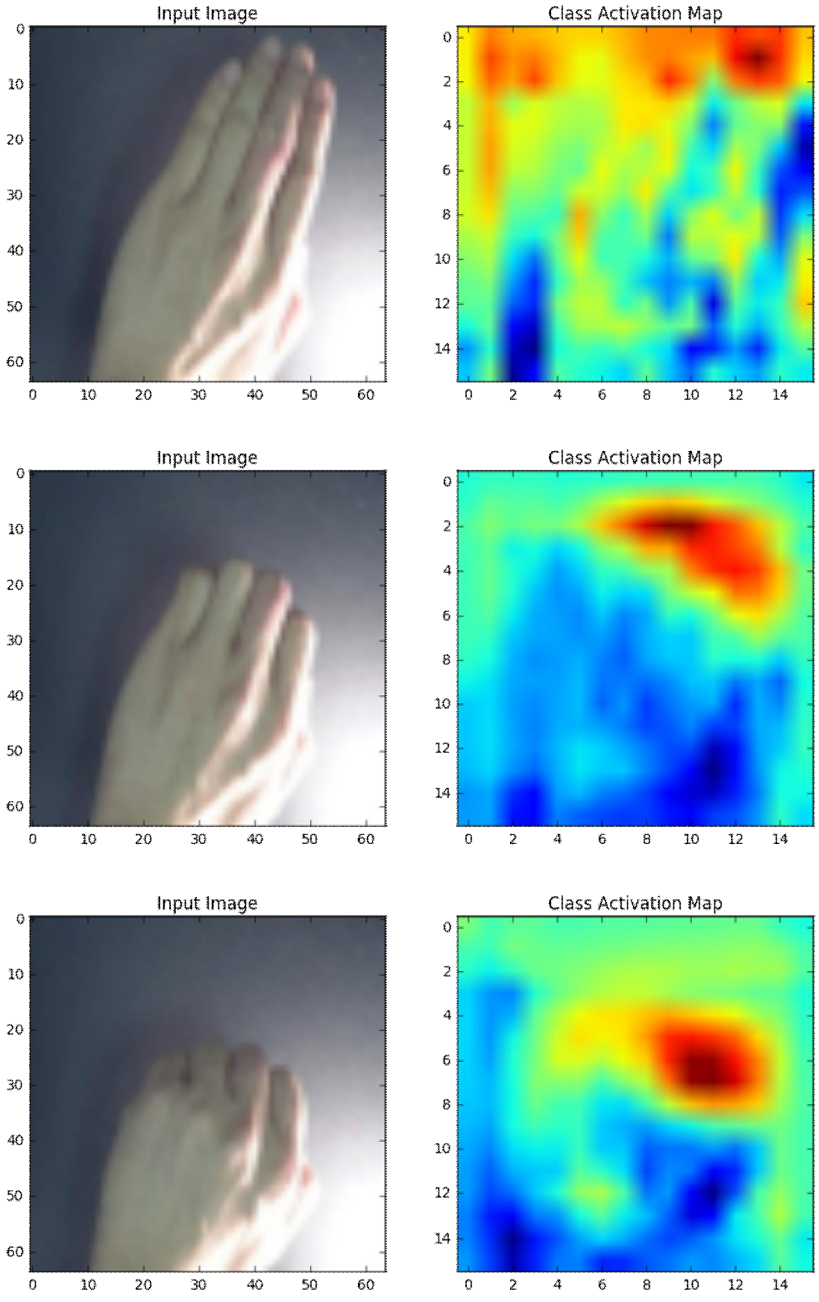
[그림 4-25] Cambridge 데이터 세트의 차영상을 이용한 활성화 맵



[그림 4-26] CAPP 데이터 세트의 원본영상을 이용한 활성화 맵



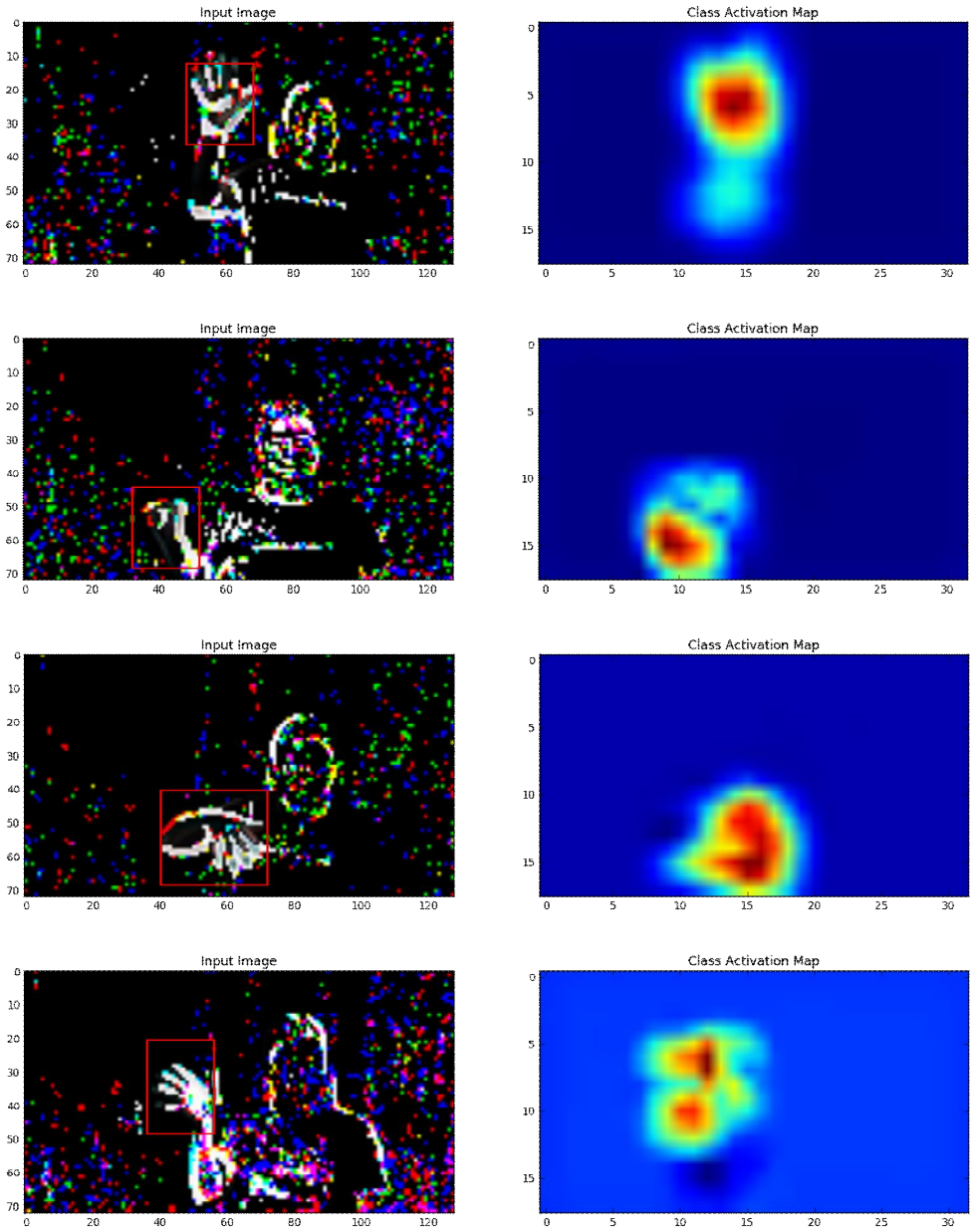
[그림 4-27] CAPP 데이터 세트의 차영상을 이용한 활성화 맵



[그림 4-28] Cambridge 데이터 세트의 활성화 맵의 다른 예

4.7 약한 지도 학습을 이용한 손 검출(Hand Detection)

객체 검출(Object detection)은 컴퓨터 비전 분야에서 많이 다루어지는 주제 중 하나로 이미지나 영상에서 원하는 객체의 위치를 정확하게 찾아내는 기술을 말한다. 앞 절에서 살펴보았듯이, 3차원 컨볼루션 신경망은 제스처를 구분하도록 학습되었지만, 클래스 활성화 맵을 이용하면 신경망이 어떤 부분을 근거로 하여 해당 클래스를 판단하였는지 시각화할 수 있고, 대부분의 제스처 판단의 근거는 손의 움직임에 있으므로 이를 이용하면 영상 중에 손의 위치를 대략적으로 찾아낼 수 있다. 이렇게 클래스를 분류하는 신경망을 학습시킨 후에 특정 클래스를 판단하기 위해 신경망이 집중한 부분을 시각화하여 객체 검출에 이용하는 학습법을 약한 지도 학습(weakly supervised learning)이라고 한다. **[그림 4-29]**에 차영상을 이용하여 학습된 3차원 신경망의 활성화 맵을 이용하여 활성화 맵의 값이 특정 값 이상인 영역을 산출해내고 그 영역에 바운딩 박스(bounding box)를 표현한 것을 나타내었다. 이렇게 본 논문에서 제안한 신경망 구조를 사용할 경우 제스처의 인식뿐만 아니라 손의 위치까지 대략적으로 파악할 수 있는 추가 효과까지 얻을 수 있다. 또한 이것은 차영상을 사용할 경우에 더욱 정확한 결과를 얻을 수 있으므로 이 또한 차영상을 이용하여 얻을 수 있는 장점이라고 할 수 있다.



[그림 4-29] 클래스 활성화 맵을 이용한 손 위치 검출

제 5 장 결 론

본 논문에서는 3차원 컨볼루션 신경망의 연산량과 파라미터 수를 효율적으로 줄이면서도 높은 손 제스처 인식률을 보이는 방법과 그 구조를 제안하였다. 첫째로 차영상을 입력으로 이용하는 방법을 제안하였는데, 이것은 영상 데이터의 특성상 이웃한 프레임 간의 픽셀값에 차이가 크지 않다는 것을 이용한 것으로, 여러가지 실험을 통하여 연산량을 줄일 수 있을 뿐만 아니라 신경망이 학습을 하는 데에 불필요한 배경을 상당부분 제거함으로서 학습 속도를 향상시킬 수 있음을 확인하였다. 두번째 방법으로 3차원 컨볼루션 신경망에 인셉션 구조를 확장하여 적용하고 또 그 필터들을 작은 구조로 분해하여 파라미터 수를 줄이는 방법을 제안하였다. 컨볼루션층에서는 전체 신경망의 90%이상의 연산이 일어나기 때문에 이러한 파라미터 수를 줄이는 것이 굉장히 중요하며, 인셉션 구조를 사용하였을 경우 일반적인 3차원 컨볼루션 신경망에 비하여 인식률도 향상되는 것을 확인할 수 있었다. 마지막으로 3차원 구조의 글로벌 평균 풀링을 사용하여 컨볼루션 신경망의 거의 대부분의 파라미터가 몰려 있는 전결합층을 최소한으로 축소하여 파라미터 저장을 위한 메모리 사용량을 효과적으로 줄일 수 있는 방법을 제안하였다. 뿐만 아니라 글로벌 평균 풀링을 사용할 경우, 시간적 공간적 변화에 강한 구조가 되기 때문에 실제 실험 결과를 통하여 제스처 인식률이 획기적으로 향상되는 것을 확인할 수

있었다.

본 논문에서는 제안한 신경망의 학습이 잘 이루어졌는지를 확인하기 위하여, 클래스 활성화 맵을 이용하였고, 이를 통해 제안된 신경망이 제스처를 인식하기 위한 특징을 잘 추출했음을 실험 결과로 확인하였다. 마지막으로, 본 논문에서 제안한 신경망 구조를 사용할 경우, 제스처 인식뿐만 아니라 손의 위치까지 파악할 수 있는 추가적인 장점이 있음을 확인할 수 있었다. 본 논문에서 제안한 방법 외에도 [14]에서 이용한 파라미터의 양자화를 통해 파라미터의 비트수를 줄이고 재학습을 하는 방식등을 추가로 적용하면, 더 적은 양의 메모리 사용량과 연산량을 가지고 효율적인 손 제스처 인식이 가능할 것으로 예상된다.

참고 문헌

[1] P. Molchanov, S. Gupta, K. Kim, J. Kautz, "Hand Gesture Recognition with 3D Convolutional Neural Networks", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Publisher, pp. 1-7, 2015.

[2] S. Ji, W. Xu, M. Yang, "3D Convolutional Neural Networks for Human Action Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 221-231, January 2013.

[3] D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, "Learning Spatiotemporal Features with 3D Convolutional Networks", IEEE International Conference on Computer Vision (ICCV), pp. 4489-4497, 2015

[4] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, J. Kautz, "Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D Convolutional Neural Network", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4207-4215, 2016.

[5] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, T. Darrell, "Long-Term Recurrent Convolutional Networks for Visual Recognition and Description", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2625-2634, 2015.

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, "Going Deeper with Convolutions", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-9, 2015.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, "Rethinking the Inception Architecture for Computer Vision", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

[8] K. Simonyan, A. Zisserman, "Two-Stream Convolutional Networks for Action Recognition in Videos", Neural Information Processing Systems(NIPS), 2014.

[9] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, F.

F. Li, "Large-scale Video Classification with Convolutional Neural Networks", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1725-1732, 2014.

[10] T.-K. Kim and R. Cipolla, "Canonical Correlation Analysis of Video Volume Tensors for Action Categorization and Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 8, pp. 1415-1428, 2009.

[11] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", Neural Information Processing Systems (NIPS), pp. 1106-1114, 2012.

[12] J. Albericio, P. Judd, T. Hetherington, T. Aamodt, N. E. Jerger, A. Moshovos, "Cnvlutin: Ineffectual-Neuron-Free Deep Neural Network Computing", ACM/IEEE International Symposium on Computer Architecture (ISCA), pp. 1-13, 2016.

[13] M. Lin, Q. Chen, and S. Yan, "Network in network", International Conference on Learning Representations, 2014.

[14] S. Zhou, Y. Wu, Z. Ni, X. Zhou, H. Wen, Y. Zou, "DoReFa-Net:

Training Low Bitwidth Convolutional Neural Networks with Low Bitwidth Gradients" , arXiv:1606.06160, 2016.

[15] P. Molchanov, S. Gupta, K. Kim, K. Pulli, "Multi-sensor system for driver's hand-gesture recognition" , IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, pp. 1-8, 2015.

[16] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization", arXiv preprint arXiv:1512.04150, 2015

[17] K. Fukushima and S. Miyake, "Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in positions", Pattern Recognition, 15:455-469, 1982

[18] Y. LeCun, L. Botto, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition", Proceeding of the IEEE, 86(11):2278-2324, 1998

[19] S. Shin, W. Sung, "Dynamic hand gesture recognition for wearable devices with low complexity recurrent neural networks",

IEEE International Symposium on Circuits and Systems(ISCAS), pp. 2274-2277, 2016

[20] Simonyan, K. & Zisserman, "A Very deep convolutional networks for large-scale image recognition", arXiv preprint arXiv:1409.1556, 2014

Abstract

3D-Convolutional Neural Network for Efficient Hand Gesture Recognition

Jin Won Lee

Department of Electrical and Computer Engineering

The Graduate School

Seoul National University

Hand gesture recognition technology refers to a technique that recognizes what kind of action it is when a person performs a predetermined action using his or her hand. This recognition technology is an important technique that can provide an effective interface for automobiles, mobile or wearable devices, household appliances, etc., since gestures do not require direct contact. A number of computer vision algorithms have been developed for hand

gesture recognition and their performance has been steadily improved. In recent years, due to the development of artificial neural network and deep learning technology, many researches have exceeded existing ones in the computer vision field. However, hand gesture recognition technology requires high complexity, large amount of computation and high memory usage. Moreover, many devices have limited computing power, so gesture recognition is still a challenging area.

This paper proposes a novel structure of artificial neural network using 3D convolutional neural network which can reduce computation and memory. Since the video data contains continuous motion in time, it is characterized by the fact that there is no big difference between two consecutive frames. Using this, a method to use the difference images between consecutive frames of RGB images that becomes from the camera as an input of artificial neural network is proposed. Inception structure, which is one of the structures that showed excellent performance by using a relatively small number of learning parameters in image classification, is extended to be applicable to video data, i.e., three-dimensional data. Factorized structure of in caption modules is also proposed. Finally, this paper proposes the 3D global average pooling instead of fully-connected layer at the end of the CNN to reduce the amount of

weight parameters.

When the proposed 3D-CNN is used, the memory usage required to store weight parameters can be reduced by about over 99% without degradation of recognition accuracy compared with the normal 3D-CNN. Also, in the case of multiplication which generates the most expensive computation, it is possible to reduce about 95% of the number of multiplications compared to the normal 3D-CNN without lowering the accuracy.

Keywords : hand gesture recognition, 3D-CNN, difference images, inception, global average pooling

Student Number : 2002-21592